(2)

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

THE DESIGN AND IMPLEMENTATION
OF A POSITION MEASURING SYSTEM FOR
A REMOTELY CONTROLLED VIDEO CAMERA

by

Peter D. Lloyd

June 1989

Thesis Advisor: Alfred W. Cooper
Co-Advisor: Hal A. Titus

Approved for public release; distribution is unlimited

Unclassified

security classification of this page

REPORT DOCUMENTATION PAGE

1a Report Security Classification Unclassified			1b Restrictive Markings
2a Security Classification Authority			3 Distribution Availability of Report: Approved for public release; distribution is unlimited.
2b Declassification Downgrading Schedule			
4 Performing Organization Report Number(s)			5 Monitoring Organization Report Number(s)
6a Name of Performing Organization: Naval Postgraduate School	6b Office Symbol (*if applicable*) 62		7a Name of Monitoring Organization: Naval Postgraduate School
6c Address (*city, state, and ZIP code*): Monterey, CA 93943-5000			7b Address (*city, state, and ZIP code*): Monterey, CA 93943-5000
8a Name of Funding Sponsoring Organization	8b Office Symbol (*if applicable*)		9 Procurement Instrument Identification Number
8c Address (*city, state, and ZIP code*)			10 Source of Funding Numbers: Program Element No / Project No / Task No / Work Unit Accession No

11 Title (*include security classification*) THE DESIGN AND IMPLEMENTATION OF A POSITION MEASURING SYSTEM FOR A REMOTELY CONTROLLED VIDEO CAMERA

12 Personal Author(s) Peter D. Lloyd

13a Type of Report	13b Time Covered	14 Date of Report (*year, month, day*)	15 Page Count
Master's Thesis	From To	June 1989	133

16 Supplementary Notation The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes			18 Subject Terms (*continue on reverse if necessary and identify by block number*)
Field	Group	Subgroup	Position measurement, microprocessor, MC68705U3, M68705EVM, optical shaft encoder, HEDS-6000,IRSTD.

19 Abstract (*continue on reverse if necessary and identify by block number*)

A position measuring system for a remotely controlled video camera was designed and built. The camera is intended to be used with the modified Advance Development Model of the AN/SAR-8 Infrared Search and Target Designation System (IRSTD) in use at the Naval Postgraduate School. The video data collected by the camera will be correlated with the infrared data from the IRSTD to develop a background data base that will be used in the developement of signal processing algorithms.

The measurement system uses two Hewlett Packard HEDS-6000 incremental optical encoders, two Motorola MC68705U3 microprocessors and two digital display devices to measure and present the camera's azimuth and elevation angles to an operator at a remote location. The azimuth can be measured over a range of 360° with a resolution of ± 0.0213° and the elevation can be measured over 24° with a resolution of ± 0.138°. The resolution is limited primarily by hysteresis, which is due to the backlash in the gears between the transducers and the axes of interest.

20 Distribution Availability of Abstract	21 Abstract Security Classification	
☒ unclassified unlimited ☐ same as report ☐ DTIC users	Unclassified	
22a Name of Responsible Individual: Alfred W. Cooper	22b Telephone (*include Area code*): (408) 646-2452	22c Office Symbol: 61Cr

DD FORM 1473,84 MAR 83 APR edition may be used until exhausted. All other editions are obsolete

security classification of this page

Unclassified

Approved for public release; distribution is unlimited.

The Design and Implementation
of a Position Measuring System
for a Remotely Controlled
Video Camera

by

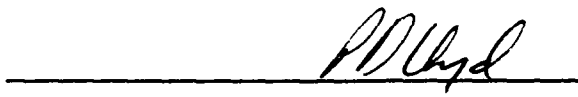
Peter D. Lloyd
Captain, United States Marine Corps
B.S., United States Naval Academy, 1979

Submitted in partial fulfillment of the
requirements for the degree of

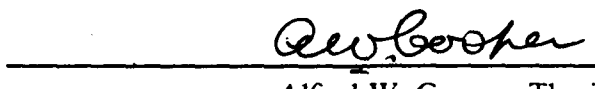
MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

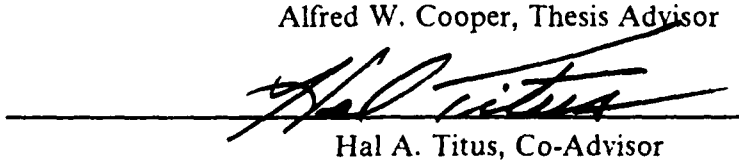
NAVAL POSTGRADUATE SCHOOL
June 1989

Author: 

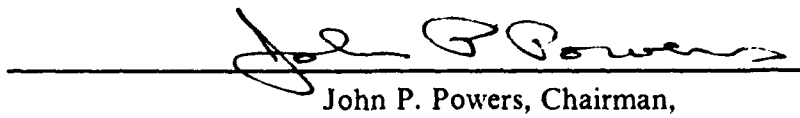
Peter D. Lloyd

Approved by: 

Alfred W. Cooper, Thesis Advisor



Hal A. Titus, Co-Advisor



John P. Powers, Chairman,
Department of Electrical and Computer Engineering

Gordon E. Schacher,
Dean of Science and Engineering

ABSTRACT

A position measuring system for a remotely controlled video camera was designed and built. The camera is intended to be used with the modified Advance Development Model of the AN/SAR-8 Infrared Search and Target Designation System (IRSTD) in use at the Naval Postgraduate School. The video data collected by the camera will be correlated with the infrared data from the IRSTD to develop a background data base that will be used in the developement of signal processing algorithms.

The measurement system uses two Hewlett Packard HEDS-6000 incremental optical encoders, two Motorola MC68705U3 microprocessors and two digital display devices to measure and present the camera's azimuth and elevation angles to an operator at a remote location. The azimuth can be measured over a range of 360° with a resolution of $\pm 0.0213°$ and the elevation can be measured over 24° with a resolution of $\pm 0.138°$. The resolution is limited primarily by hysteresis, which is due to the backlash in the gears between the transducers and the axes of interest.

THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

This thesis is prepared in conjunction with research sponsored in part by the Naval Sea Systems Command, PMS-421, and by the Naval Postgraduate School.

I would like to take a moment to thank the following people who have offered so much and so generously as I have undertaken this project:

My parents who have always helped me through the hard times.

Professor A. W. Cooper for his confidence and insight.

Professor Hal Titus for his constant encouragement, guidance and tremendous sense of humor which made even the difficult times enjoyable.

John Glenn, Jerry Lentz and Eric Moore for their continued technical assistance.

ET2 Akerson for his dedicated work in designing and building the printed circuit boards.

George Jaksha for his expert advice and work in modifying the camera servo mechanism.

I offer my sincere thanks to my fellow officers and students, especially my good friends Capt Steve Spehn and CPT Ken Cockerham for their friendship and thoughtful suggestions.

Finally, I offer my thanks and love to my dear wife, Ellen, who has given so very much in the past two years.

I. INTRODUCTION

A. BACKGROUND

The old saying that "every solution breeds new problems", while somewhat pessimistic, quite often rings true in today's increasingly technical world. One such example is the use of infrared (IR) sensors for the detection, tracking and/or identification of targets in a combat environment. IR sensors are ideally suited for use on today's battlefield. They are passive, i.e., they do not need to emit energy in order to detect the presence of potential targets. This allows them to operate during times of emission control when many other target detection systems are useless. They have the ability to "see through" many forms of camouflage and concealment, dust, clouds, smoke, etc., that might otherwise afford an enemy target a safe haven. Additionally, because almost everything on today's battlefield generates some degree of infrared energy, IR sensors can be used to locate and identify a wide variety of targets.

The extent to which a particular IR sensor is useful depends primarily on its ability to detect and identify targets reliably and accurately. Detection of the target is primarily a function of the IR sensor's sensitivity. The classification of a received IR signal as a potential target or as background noise, while still dependant on the sensitivity of the sensors, is primarily a function of the quality of the signal processing algorithms being used to process the received signals. In addition to being reliable these algorithms must be able to process the received signals in "real time" if the system is to be an effective weapons system.

Creation of a background data base that can be used to test some of these algorithms has been one of the tasks being performed by the Naval Academic Center for Infrared Technology (NACIT) located at the Naval Postgraduate School (NPS). The Advanced Development Model (ADM) of the AN/SAR-8 was sent to the NACIT in January of 1984 from the Naval Surface Weapons Center (NSWC) at Dahlgren, Virginia. The ADM was modified, calibrated and placed in service at NPS in December, 1987. The modified version of the ADM, the Infrared Search and Target Designation (IRSTD) System, is currently operational at NPS. [Ref. 1: pp. 8-12]

One way to enhance the usefulness of the IR data being collected at NPS would be to collect video data concurrently with the IR data. A video image of a portion of the horizon would permit visual identification of IR sources in that region. This additional

tional information could be an aid in the development of the signal processing algorithms for the IRSTD. Accordingly, a decision to proceed with video data collection was made by NACIT, and a camera system was purchased. Components of the system include;

- RCA (TC1005/01), Closed circuit video camera.
- PELCO (AI700), Automatic iris servo.
- PELCO (F1.5X), 1.5 times range extender.
- PELCO (MLZ6DT), Desk top lens remote control module.
- PELCO (PT1250DC), Heavy duty Pan/Tilt servo.
- PELCO (MPTV1510DT), Pan/Tilt remote control unit.
- Panasonic (WV-5410), Video Monitor.

In Ref. 1 Ayers describes the IRSTD's detectors:

> The IR detectors consist of two vertical arrays of sensing elements in the focal plane of the Schmidt telescope. The telescope is rotated so as to sweep the image across the detector arrays. Each array incorporates a column of 90 indium antimonide photovoltic linear detector elements. These two arrays are independent of each other and are covered by filters which pass selected wavebands of IR radiation in the 3 to 5 micrometer range. Each element has the angular dimensions of 2 X .3 milliradians with the larger being its height. Designated as the lead and the lag, these two arrays are separated by about one-half degree in azimuth. [Ref. 1: p. 17]

Thus, as the IRSTD scans the horizon the resulting IR image has a resolution of approximately 10^{-4} radians (0.00573°) in the horizontal plane and 0.23° in the vertical plane. The video system's smaller field of view can be remotely controlled using the control units listed above. Thus, for the video data to be of any use in the development of the signal processing algorithms, the camera's orientation must first be known, and, in order to determine the pixel-to-pixel correlation between the IR image and the video image, the position of the video camera needs to be known with the same kind of accuracy as the IR image. The design and implementation of a position measuring system for this remotely controlled video camera is the subject of this thesis.

B. DESIGN SPECIFICATIONS

The design specifications for this problem were relatively straight forward. The position measuring system needed to meet the following criteria.

- The system should be able to measure the elevation angle (tilt), above and below the horizontal reference plane of the camera over a range of $\pm$ 12°.
- The measured elevation angle should be accurate to within $\pm$ 0.23° .

- The system should be able to measure the bearing (pan), left or right of some arbitrary reference, of the camera over a range of 360°.
- The measured bearing should be accurate to within $\pm 10^{-4}$ radians.
- The output should be displayed in a convenient form. The display should be collocated with the camera servo remote controls, an indoor site approximately 200 meters away from the camera.
- Portions of the measuring system required to be collocated with the camera should be weatherproofed.
- The system must be reliable and should be simple to operate.

II. DESIGN STRATEGY

A. GENERAL

A position measuring system, like any system, is a combination of devices interconnected to perform a certain function. The most basic position measuring system (see Figure 1) consists of only four such devices: a transducer, a signal conditioner, a display device and a power supply. More complicated position measuring systems include those designed to take a number of different measurements either simultaneously or consecutively. Still more complex systems multiplex these various measurements over a single channel to some distant location where they can be processed and displayed. [Ref. 2: pp. 2-14]

The design specifications for the camera position measurement system (subsequently referred to as the "measurement system") required that two measurements, pan and tilt, be taken simultaneously and transmitted some distance to a remote display. Two separate transducers, capable of independent operation, were therefore required. In order for both the azimuth and the elevation to be displayed simultaneously two display devices were also required.

Several pairs of RG-178 coaxial cable were available to transmit signals between the camera servo and the remote control site. Since the cable was available and it was desirable to reduce the system complexity, a decision was made not to multiplex the data over a single channel. Instead, each measurand would have a separate transducer, a dedicated signal processor and a unique display device. Position information for each axis would be transmitted over a dedicated channel.

The physical locations of the transducers and the display devices were dictated by the design specifications; however, there was some flexibility in deciding where to locate the signal conditioner. Site selection was based on an attempt to maximize total system performance and simplicity while ensuring the maintainability and environmental integrity of the signal conditioner. The only advantage to locating the signal conditioner with the camera servo and the transducer would have been to limit the distance that the transducer's output signal would have to be transmitted to the processor. On the other hand, separating the signal conditioner and the transducers would limit the distance over which the conditioned signal would have to be transmitted to the display device. The trade-off here was not clear cut and would probably depend on the specific hardware

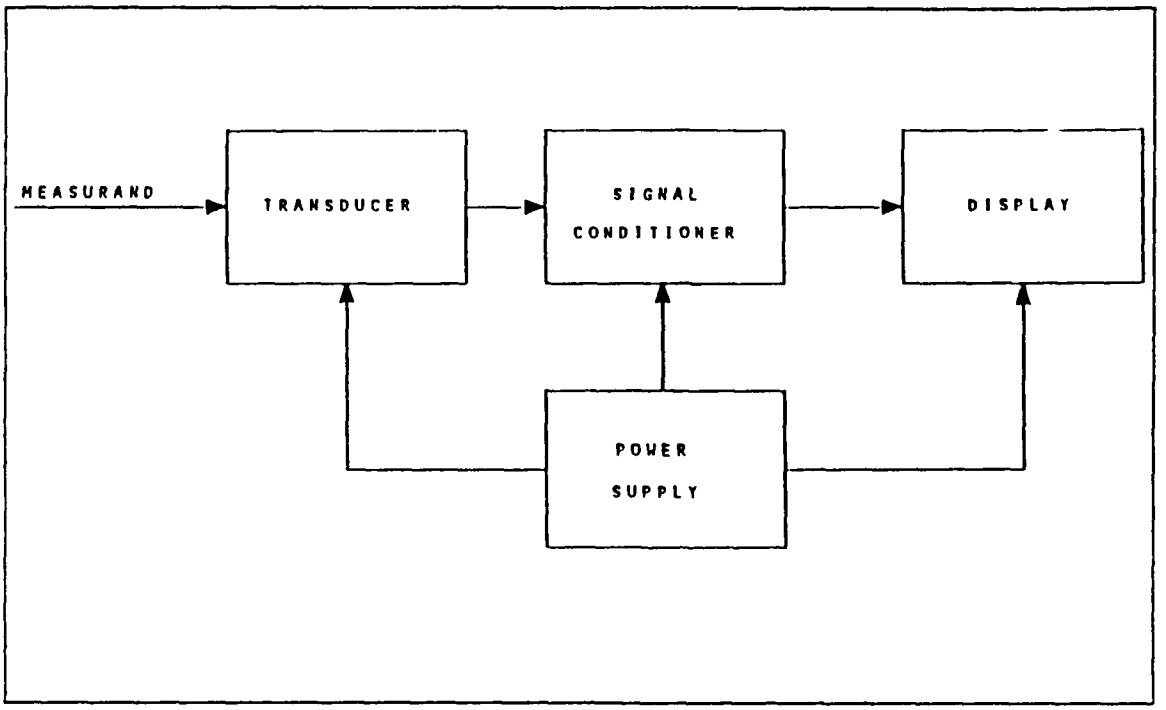


Figure 1. Basic Electronic Position Measuring System: From Ref. 2: p. 2

used and the speed of rotation of the camera servo. There was one significant advantage, however, to collocating the display and the signal processor; since they would both be indoors, the need for weatherproofing the signal processor would be eliminated.

Power was available at both ends of the system, there was therefore no requirement to have a common power supply for the entire system. One supply could be used to provide power to the two transducers, collocated with the camera servo, and a second supply could power the signal conditioner and the displays.

A block diagram of the prototype measurement system is shown interconnected with the camera positioning system in Figure 2. Once this basic system layout had been determined, proper selection of the actual hardware was necessary. The design criteria were the primary consideration in the initial stages of the hardware selection. Final selection of the specific components, however, involved balancing additional factors, such as availability and cost against the system requirements.

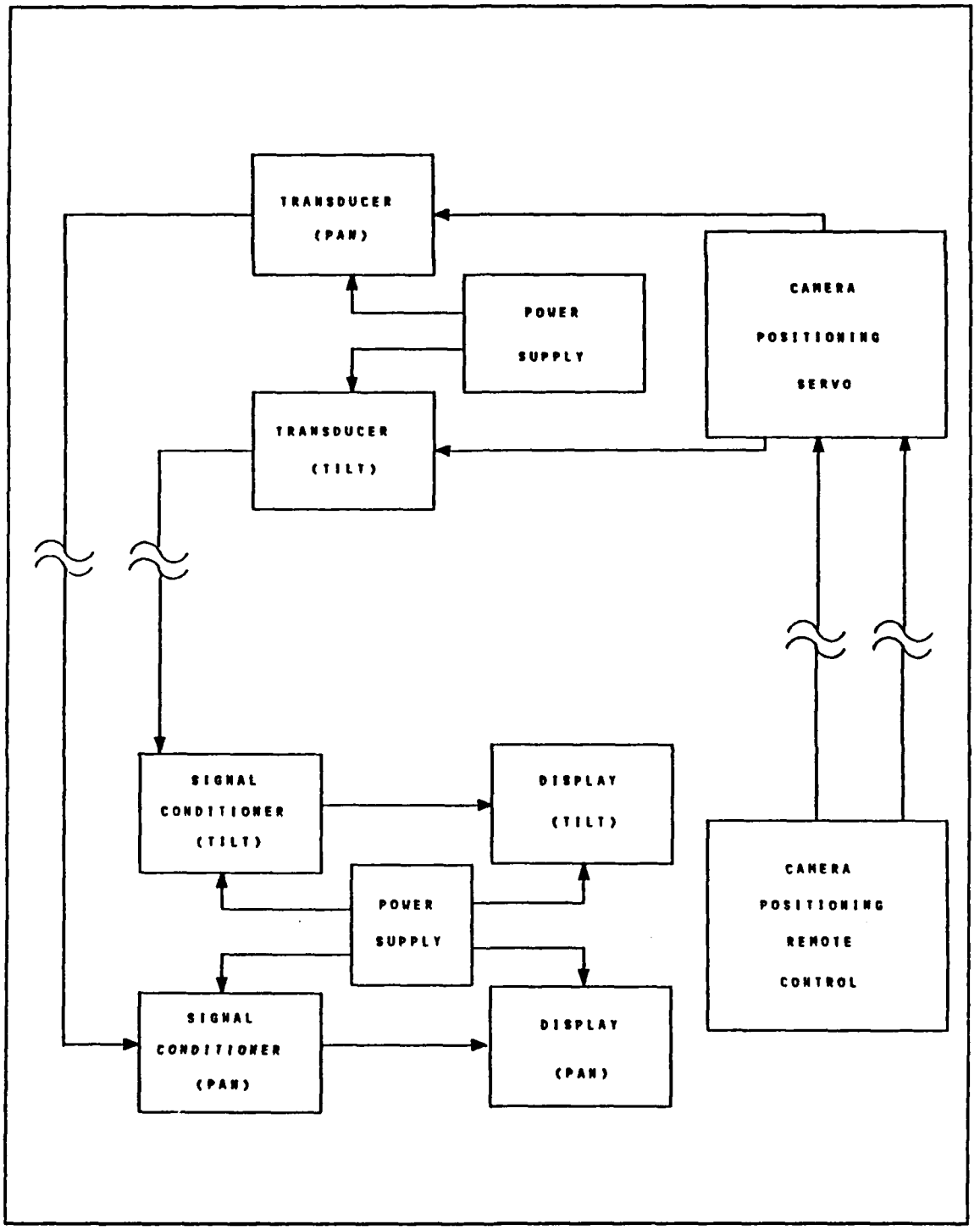


Figure 2. Camera Position Measuring System

B. TRANSDUCERS

1. General

Since in practice most measurement systems do not have the ability to respond directly to the measurand, transducers are used to convert one physical quantity (e.g., angular position) into another, more usable quantity or signal (e.g., an electrical signal) [Ref. 3: p. 1-4]. The transducer is therefore a vital part of any measurement system, and although none of the components of this system could have been chosen independently of the other elements, proper transducer selection appeared to be the key to meeting the design specifications. Thus, selection of a transducer was the next step in the design process.

Using the selection guidelines given by Norton on pages 51-53 of Ref. 2 and the design criteria stated previously, several observations and decisions were made which significantly reduced the number of transducers considered feasible for use in the measurement system. The fact that the servo was capable of rotating the camera left or right, and up or down suggested that the transducer should be capable of detecting both increasing and decreasing angles on both axes. Additionally, since the servo was anticipated to rotate the camera through one 360° arc in the horizontal plane, the transducer used to measure this angle (subsequently referred to as the "pan transducer") needed to have a comparable range capability. The range requirement for the "tilt transducer" (used to measure the elevation angle) was much less restrictive. These factors, angular bidirectional capability and full scale range, eliminated a great number of transducers from the list of candidates.

The list was further narrowed by the accuracy requirements previously specified. Again the accuracy in the horizontal plane placed a much more severe limitation on the selection of a transducer than the accuracy requirements for the vertical plane. The following paragraphs in this section outline the logic used in the selection of the transducers. The factors considered in the selection process included the following;

- The accuracy requirements given in the specifications.
- The ease with which a specific transducer could be installed on the servo.
- The ability to weatherproof the servo and the transducer once the transducer was installed.
- The rotation speed of the camera about the servo axes.
- The extent to which a particular transducer/mounting configuration would modify the measurand.
- Cost effectiveness.

- Availability.
- Signal conditioning requirements.
- The extent to which the selection of a particular transducer would simplify or complicate the modification or expansion of the measurement system.

2. Transducer technologies

Displacement transducer technologies fall into three very broad categories depending on whether they are capable of measuring linear or angular displacement, or both. Some technologies could be eliminated immediately since they were clearly not suited for measuring angular position. Strain gauge displacement transducers, inductive displacement transducers and vibrating-wire displacement transducers are examples of such devices. The following paragraphs briefly describe the different transducer technologies which were investigated. Table 1 on page 18 summarizes the salient points of the discussion. [Ref. 2: pp. 90-117].

a. Reluctive Displacement Transducers

The rotary variable differential transformer (RVDT), which operates by detecting a change in the reluctance between coils, offers excellent resolution, dynamic characteristics, linearity and life expectancy [Ref.3: p. (2-14)]. Figure 3 shows a schematic diagram and a simplified cross-section of an RVDT. The ferromagnetic, cardioid-shaped core is attached to a shaft as shown. As the shaft rotates, the inductive coupling between the primary and each of the secondary coils changes. When the cam is symmetric with respect to the two secondary coils, their output voltages are equal but opposite in phase which results in a differential output voltage of zero. As the shaft rotates away from this "null" the differential output voltage varies as shown in Figure 4. The linear region of the curve is limited to the angles between $\pm 40°$ of the reference. Thus, the RVDT could not be used on the Pan axis. It was however, initially considered as a candidate for the Tilt transducer. [Refs. 2: pp. 93-99, 3: pp. (9-10)-(9-13) and 4: p. 19]

Another type of reluctive displacement transducer is manufactured by Farrand Controls. Their INDUCTOSYN rotary position transducers have accuracies to $\pm$ 1.5 arc sec ($\simeq(4 \times 10^{-4})°$). Unfortunately, these devices are 11.89 inches in diameter, and mounting them on the camera servo would have been extremely difficult, if not impossible. [Refs. 2: pp. 89-111, 5]

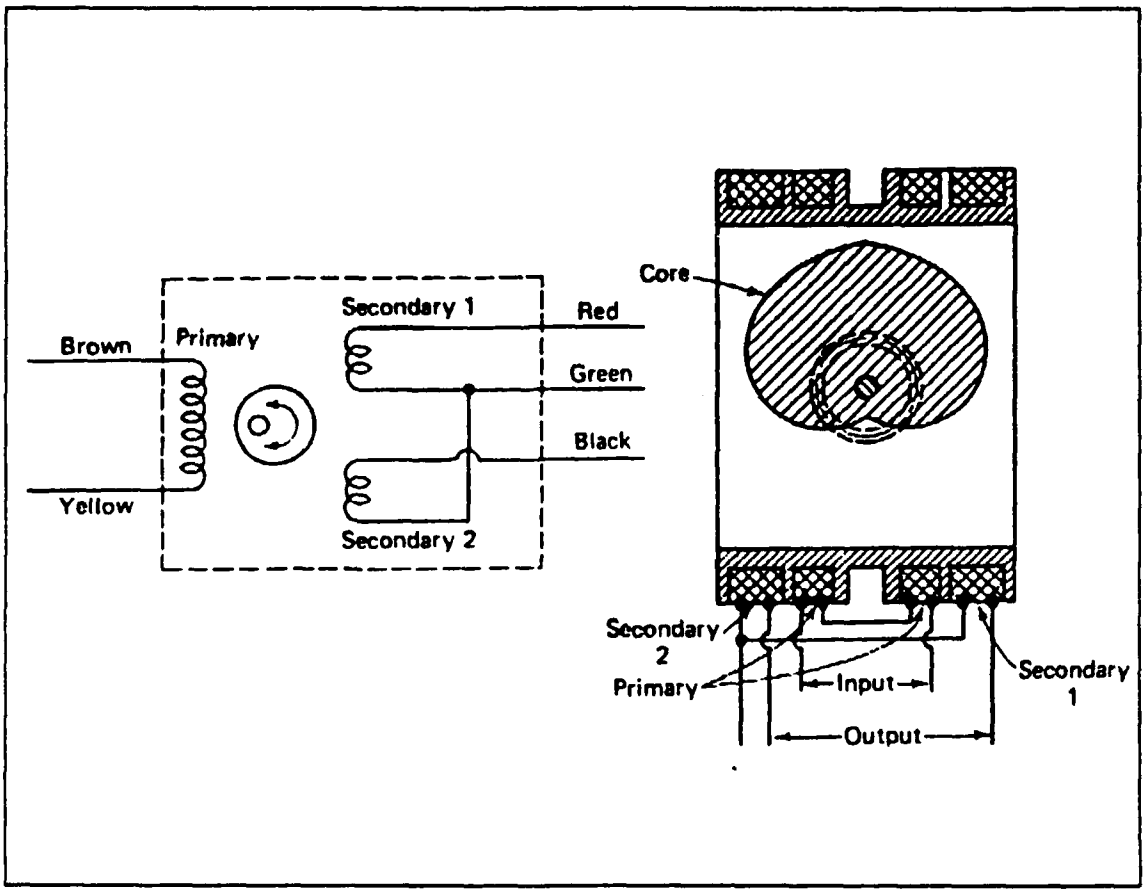


Figure 3. RVDT Schematic: From Ref. 2: p. 98.

b. Capacitive Displacement Transducers

Angular displacement can also be measured by coupling the rotating component to the shaft a of variable capacitor in the manner shown in Figure 5. Lenk describes the operation of the capacitive displacement transducer quite succinctly.

> The capacitor ... consists of a metal plate that moves between two fixed metal plates as a shaft is rotated. The three plates, and the air between them, form a capacitor with a capacitance that varies in proportion to the degree to which the plates are meshed. When the plates are completely meshed, the capacitance is at its maximum. When the plates are completely unmeshed, the capacitance is at minimum. [Ref. 4: p. 18]

Capacitive displacement transducers offer many of the advantages of reluctance displacement transducers. They have an effective range of about 300° which, while better than the RVDT's range, is still not adequate for the pan axis [Ref. 4: p. 18]. Additionally they are more sensitive to changes in the ambient temperature. Since the

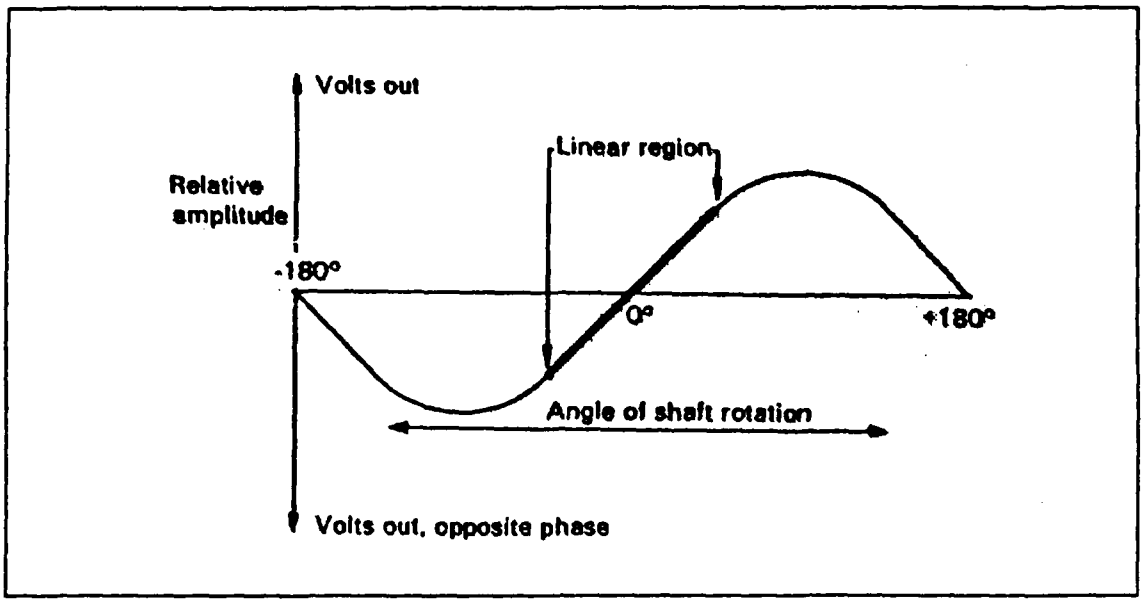


Figure 4. RVDT Output Characteristics: From Ref. 2: p. 95.

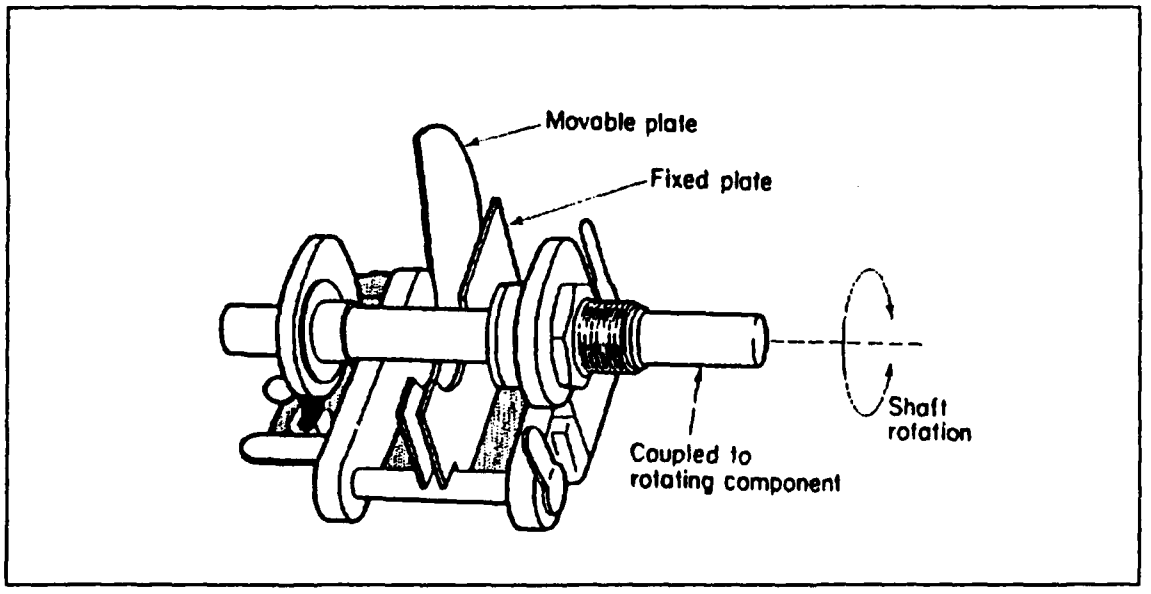


Figure 5. Capacitive Displacement Transducer: Ref. 4: p. 18.

measurement system was being designed to be used outdoors, the use of a capacitive displacement transducer would have required the addition of a temperature compensator in the design. While possible, this would have increased the system complexity considerably. [Ref. 2: pp. 90-91]

c. Potentiometric Displacement Transducers

Another very common and relatively simple family of transducers operate by measuring the change in resistance caused by a change in the measurand. There are a wide variety of such devices available. The basic form of potentiometric angular displacement transducers uses a resistance element, formed into an arc, and a movable electrical contact that rotates about the axis of interest. By measuring the change in resistance that results from a change in position, one is able to determine the angular displacement. The resistive element is typically a wirewound element, the resolution of which is determined by the number of turns per unit length of the resistance element. The angular resolution can be increased by increasing the turn density (wires/degree) of the potentiometer.

The practical limit for wire spacing on wirewound elements according to Ref. 6 is between 500 and 1000 turns per inch. From Figure 6 one can see that this limits the angular resolution for a single-turn device to

$$\Delta\theta(\text{rad}) \simeq \tan\left(\frac{\Delta x\,(\text{in})}{R(\text{in})}\right) \tag{1}$$

Therefore,

$$\Delta\theta(\text{rad}) \simeq \frac{2\Delta x}{D} = \frac{0.002(\text{in})}{D(\text{in})} \quad . \tag{2}$$

To achieve the 10^{-4} rad resolution, specified for the Pan axis, with a single turn potentiometer would therefore require a 20 in diameter potentiometer. Mounting a device this large on the camera servo was simply not feasible. Potentiometers are, however, available with multiple turns. Shaped in a helix fashion similar to that shown in Figure 7, the total length of the potentiometer can be increased, which in turn increases its resolution, without increasing the diameter of the device.

Increasing the resolution of the potentiometer by increasing the turn density in any of the manners described above, however, increases the output impedance of the device, which leads to increasing nonlinearity between the measurand and the transducer

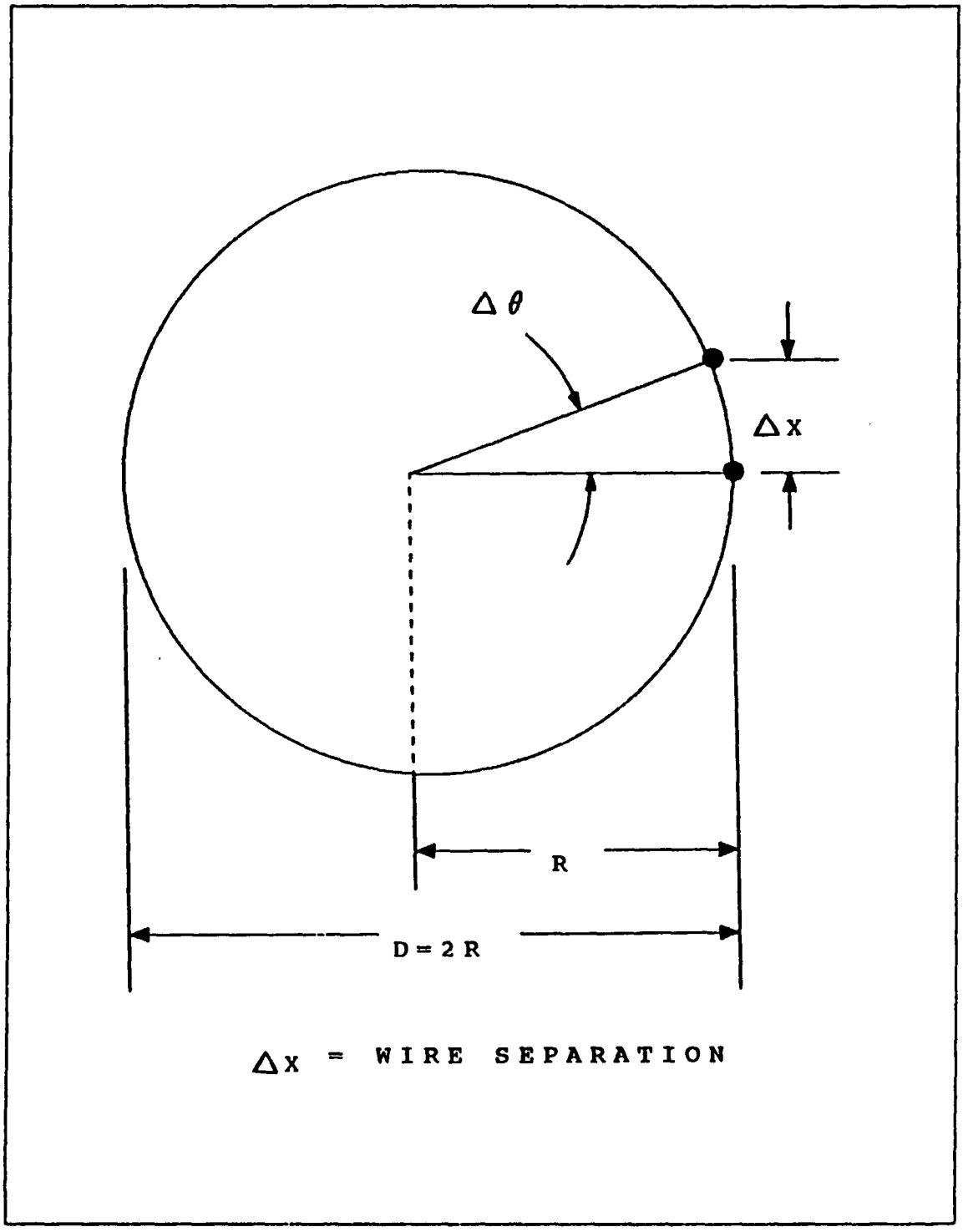


Figure 6. Geometry of an Angular Potentiometric Transducer

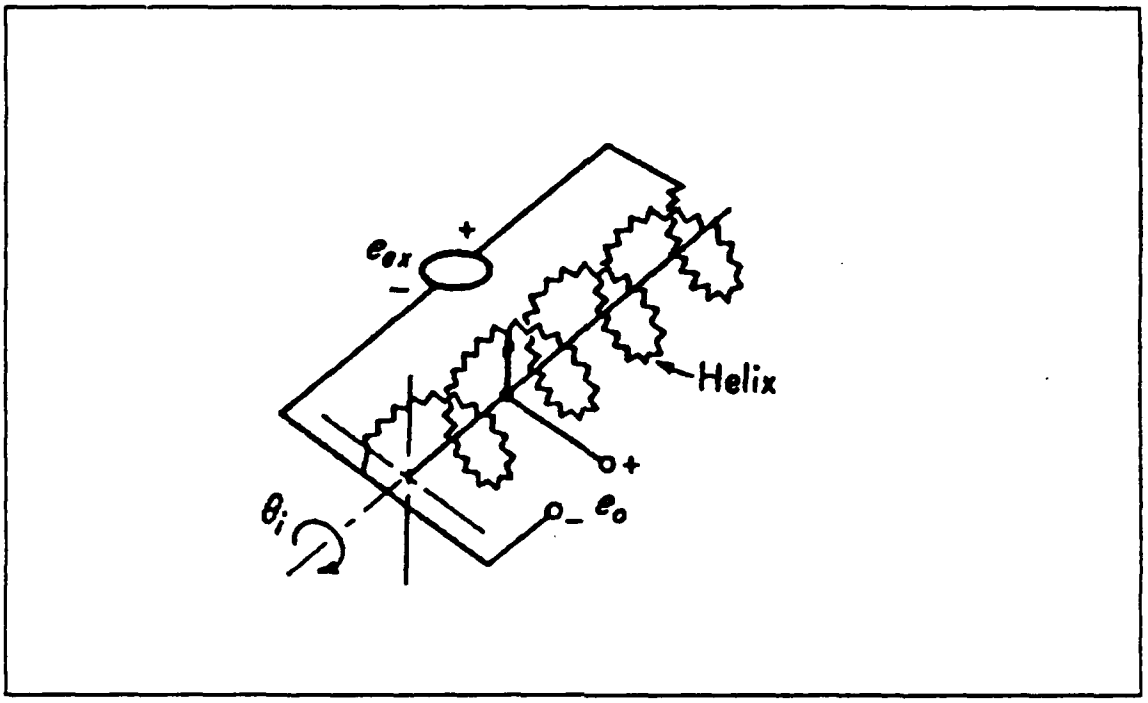


Figure 7. Multiturn Potentiometer: From Ref. 6: p. 218

output. The nonlinear relationship can be seen by analyzing the circuit shown in Figure 8. In the diagram the following variable definitions apply:

- e_X = Input voltage,
- e_o = Output voltage,
- R_p = Total resistance of the potentiometer,
- R_m = Meter resistance,
- x_t = Total range of the potentiometer, and
- x_i = Actual displacement of the potentiometer.

Assuming that R_p is uniformly distributed over x_t, analysis of the voltage divider circuit gives,

$$\frac{e_o}{e_X} = \frac{1}{\frac{x_t}{x_i} + \left(\frac{R_p}{R_m}\right)\left(1 - \frac{x_t}{x_t}\right)} \quad . \tag{3}$$

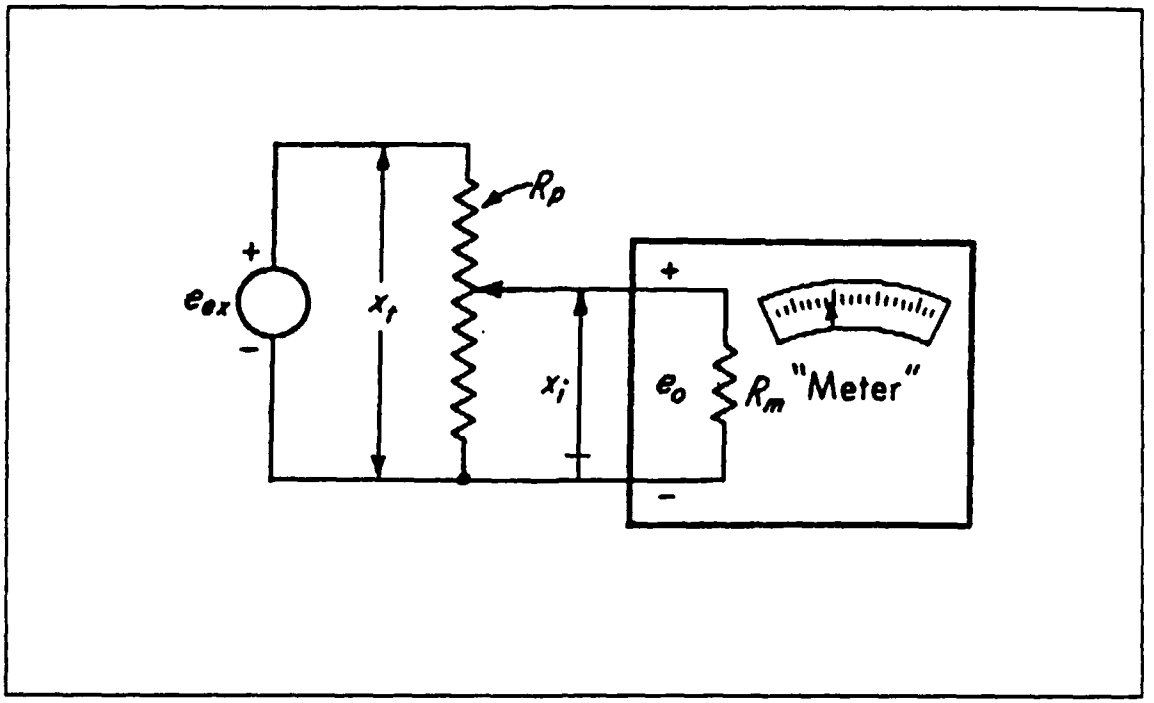


Figure 8. Potentiometric Transducer: From Ref. 6: p. 219

Thus, the ideal (i.e., linear) relationship,

$$\frac{e_o}{e_x} = \frac{x_i}{x_t} \quad , \tag{4}$$

is true only when $R_p / R_m = 0$, and since $R_p \neq 0$, and $R_m \neq \infty$, the nonlinear relationship in (3) will always exist. Doebelin states that "for values of $R_p/R_m < 0.1$ the position of maximum error occurs in the neighborhood of $x_i/x_t = 0.67$, and the maximum error is approximately $15(R_p/R_m)$ percent of full scale." [Ref. 6: p. 218] Other potentiometric transducers, which use a resistive element made of carbon film or a conducting plastic, are not subject to the same kind of resolution limitations as wirewound devices; however, they do have high output impedances and the corresponding nonlinearities described above. [Ref. 6: pp. 217-224]

A high quality multiturn potentiometer used in conjunction with a high quality voltmeter offered one possible solution to the design problem. However, the nonlinearity of this arrangement was a significant disadvantage, and the primary reason why potentiometers were not used.

d. Encoders

The angular displacement transducer, referred to in general as an angular encoder or shaft encoder, converts an angular displacement into a digital signal without the use of an analog-to-digital converter. In today's increasingly digital world this can be a distinct advantage.

There are three different transduction methods used in encoders. Magnetic encoders use a pattern made from magnetized and nonmagnetized segments and one or more magnetic sensors that register as either "1's" or "0's" depending on the magnetic characteristics of the section that they are adjacent to. Brush-type encoders are similar, but the sections are made of conducting and nonconducting materials. The conductors are all tied to a common source and the "sensor" is one or more brushes connected to the output. When the brush is in contact with one of the conductors the output is "on" and when the brush is in contact with an insulator the output is "off". Optical encoders (See Figure 9) use a pattern of opaque sections marked on an otherwise transparent disk. A light emitting diode (LED), or other light source, is placed on one side of the disk, and as the disk rotates a light sensor on the other side of the disk "sees" periods of dark and light which it converts into a digital signal. [Ref. 2: p. 106]

Angular encoders are further categorized as either absolute encoders or incremental encoders. Absolute encoders, similar to the optical encoder shown in Figure 9, use a multitrack pattern on a code wheel to produce a unique coded output signal for each incremental change in the measurand. These wheels use a variety of codes, including binary code, binary-coded decimal (BCD) and Gray code, to determine the shaft position. The resolution of an absolute shaft encoder is limited by the number of tracks on the disk and the type of code used. A simple binary or Gray code encoder, which are more efficient than the BCD encoders, with N tracks has an optimum angular resolution of

$$\Delta\theta = \frac{360^\circ}{2^N} \quad . \tag{5}$$

Thus, to achieve the desired resolution of 10^{-4} radians on the Pan axis with an absolute encoder would require a code wheel with $N \geq 16$. Since the output from the encoder is unique for each position, these devices are not affected by power outages, and the requirements for a signal processor for such a device would be limited to a simple decoding circuit. These encoders can measure angles of up to 360° without ambiguity.

Incremental encoders use a code wheel which has only one track. As the shaft of the incremental encoder rotates the output from the encoder is a series of equally spaced pulses. These pulses can then be used as an input signal to an up/down (U/D) counter of some sort. The output from the counter is an indication of the displacement of the axis from some predetermined reference. The resolution of an incremental encoder is a function of the number of pulses the code wheel generates per revolution and is given by;

$$\Delta\theta = \frac{360^\circ}{n \times PPR} \tag{6}$$

where PPR is the number of pulses per revolution of the code wheel, and n is the number of revolutions that the code wheel makes per revolution of the axis of interest. If the incremental shaft encoder was mounted on the shaft of interest (n = 1) at least 62,832 PPR would be required to ensure a resolution of 10^{-4} radians.

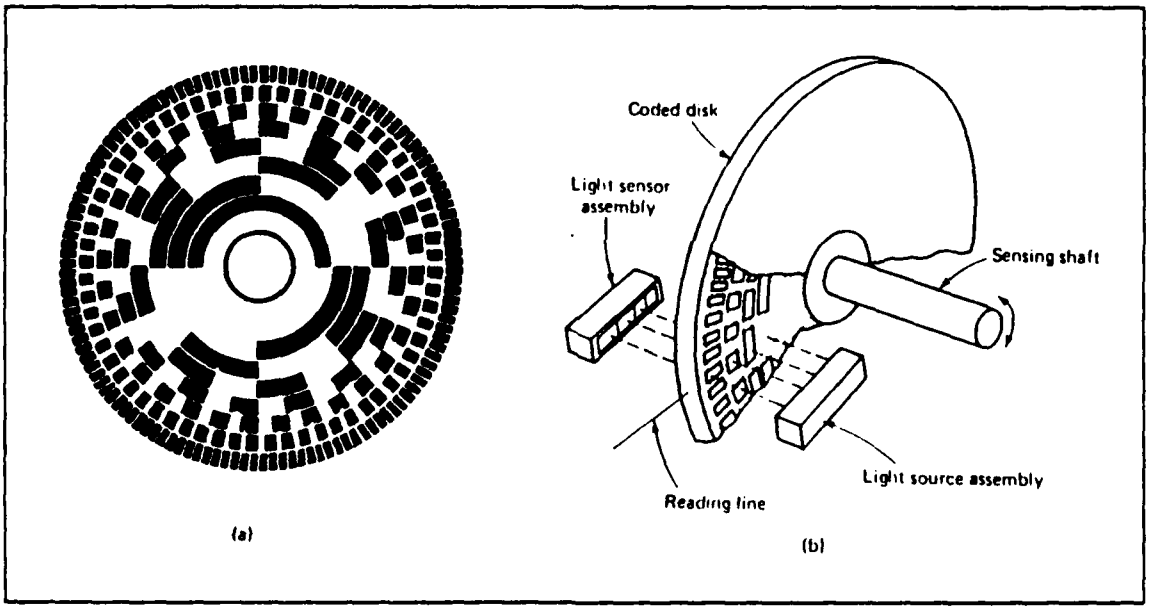


Figure 9. Absolute Photoelectric Angular Encoder: (a) typical code disk; (b) encoder elements. From Ref. 2: p. 107.

Shaft encoders currently range in price from less than $100.00 to several thousand dollars depending on their capabilities and the method of transduction used. Litton Encoder's Model 60 absolute shaft encoder uses a natural binary code, has 15 tracks (0.1917 mrad resolution) and is available "off the shelf" for approximately $3000.

Other absolute encoders with comparable resolutions are available at similar prices. No incremental encoders were found which offered the same kind of resolution as Litton's Model 60; however, because they are not limited to one revolution of the axis, the resolution of an incremental encoder can be improved by a factor of n by causing its code wheel to rotate n times for every rotation of the axis of interest. Incremental encoders with 1024 PPR are available from a variety of manufacturers for about $100 each. Connecting such a device to the axis of interest via a gear train with a 50:1 ratio would theoretically result (from (6)) in a resolution of 0.1227 mrad. Incremental encoders do require more complex signal processing than absolute encoders, and they are affected by power shut-off. Additionally, unlike absolute encoders, any missed or erroneous count that occurs with an incremental encoder will cause a persistent error. [Ref. 7: p. 16]

After considering the measurement system performance criteria and the capabilities and limitations of the various transducers, the use of an incremental, optical shaft encoder appeared to be the best selection for the transducer for each axis. This decision involved balancing the various advantages and disadvantages of the different transducers. The following list gives a summary of the key considerations in this decision.

- Small, lightweight, highly accurate and relatively inexpensive models were readily available. Being small and lightweight suggested that mounting and weatherproofing the transducers in the camera servo should not be too difficult.
- Absolute encoders offer almost all of the advantages of the incremental encoders; they are not affected by power outage, they require less complex signal processing and one time counting errors do not persist. However, these features did not seem to justify the additional price of an absolute encoder.
- Linearity and loading problems associated with the potentiometric displacement transducers were avoided.
- Direct conversion of the measurand into a digital signal precluded the requirement for an A/D converter. (This would have been a disadvantage had the use of an analog signal conditioner been anticipated.)
- Using two identical transducers, each capable of meeting the specifications for the pan axis, would reduce system complexity while still ensuring that the design specifications were met.

C. MOUNTING THE TRANSDUCER

Once the decision to use an incremental optical shaft encoder was made, selection of a specific model remained. Before selection of an actual piece of hardware could be

Table 1. ANGULAR DISPLACEMENT TRANSDUCERS

TRANSDUCER DESIGN	RANGE	RESOLUTION	LINEARITY	OTHER
Reluctive Displacement (RVDT)	0° –360°	Theoretically infinite; Limited by the signal conditioner.	Poor beyond ± 40°	
Reluctive Displacement (Farrand's INDUCTOSYN)	0° –360°	$\pm (4 \times 10^{-4})°$	Good	Large size limits usefulness with the camera measuring system.
Capacitive Displacement	0° –300°	Theoretically infinite; Limited by the signal conditioner.	Good	Temperature sensitive
Potentiometric Displacement	0° –3500°	Device dependent	Device dependent	The trade-off between range/resolution and linearity due to the loading effect of the noninfinite impedance of the signal conditioning devices used.
Absolute Encoders	0° –360°	$\frac{360°}{2^N}$	Good	No A/D conversion required. Insensitive to power shut-off.
Incremental Encoders	$\pm \infty$	$\frac{360°}{n \times PPR}$	Good	Simple. Requires more signal processing than absolute encoder but is less expensive.

done though, one additional practical consideration had to be made; where and how could a transducer be mounted in or on the servo in order to measure the position of the Pan and Tilt axes?

The physical layout of the camera positioning servo made direct connection of any type of transducer to the axes of interest virtually impossible without major modification of the servo itself. Major modification of the servo would have been expensive, time consuming and outside the scope of this thesis. It was not considered an option in this case. Measurement of the Pan and Tilt axes' displacements was most readily accomplished indirectly. Each axis of the servo is positioned by a separate dc motor via a gear train. By mounting the shaft encoder code wheels to the sprockets (items 42 and 43 in

Figure 10) which are each attached to one of the worms, an indirect measurement of the position of each of the wormgears (items 5 and 7 in Figure 10) was possible. This approach, made necessary by the servo design, was a mixed blessing.

The backlash in a worm-wormgear connection will cause the position of the worm to be different for any given wormgear position, depending on whether that position is approached from a clockwise or a counterclockwise direction. In order to correctly determine the displacement of the wormgear by measuring the displacement of the worm, the amount of backlash present must be known (i.e., would have to be determined experimentally) and accounted for by the measurement system. This nonlinear source of error would not have been a concern if the servo was only required to rotate in one direction. This was not the case, however, and hysteresis eventually was determined to be the largest single source of error in the measurement system. A more complete discussion of this topic is included in Chapter IV of this thesis.

Assuming for the moment that the effects of the backlash in the gears could have been completely compensated for, mounting the shaft encoders on the worm provided a measurement advantage analogous to the mechanical advantage afforded by the gear train. Initial measurements indicated that each of the worms turned through 18,000° ($50 \times 360°$) for every 360° rotation of the corresponding wormgear. This meant that a shaft encoder with 100 divisions per 360° attached to the worm axis could do the same job as a 5,000 division per 360° encoder attached to the wormgear axis.

D. SELECTING AN OPTICAL SHAFT ENCODER

Once the basic decisions to use incremental optical shaft encoders and to mount the encoders on the worms inside the servo housing had been made, selection of the specific pieces of hardware was relatively straight forward and was primarily a matter of convenience and expediency.

Returning to the design specifications for a moment; the most stringent requirement was to be able to measure the position of the pan axis to within $\pm (5.73 \times 10^{-3})°$. To determine the required resolution for the optical shaft encoder, the following calculations were performed.

First,

$$\frac{360°/\text{Revolution}}{5.73 \times 10^{-3}°/\text{Division}} \simeq 62{,}827 \frac{\text{Divisions}}{\text{Revolution}} \quad . \tag{7}$$

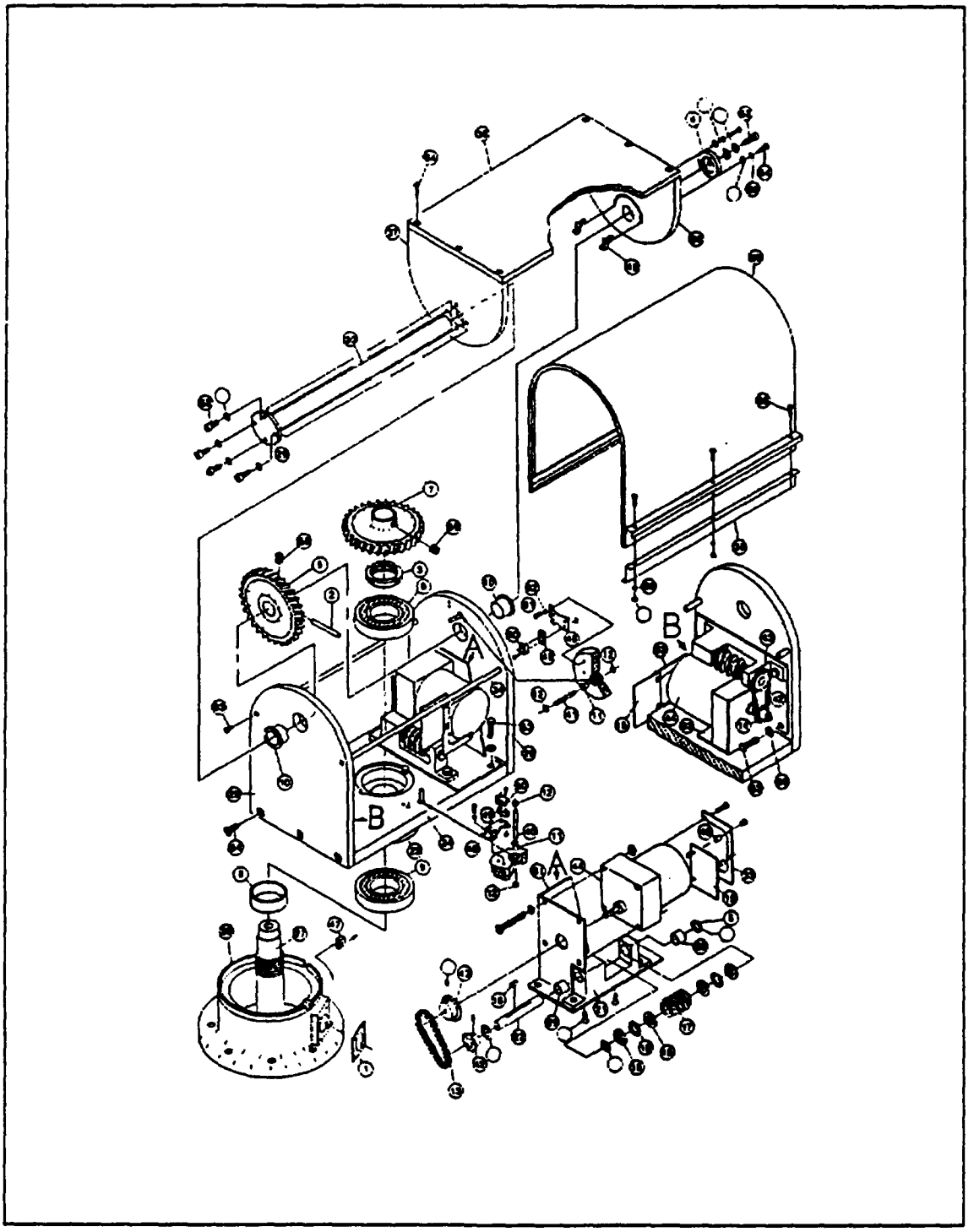


Figure 10. Camera Servo: From Ref. 8: p. 12.

Then, considering the 50:1 gear advantage,

$$\frac{62.827}{50} \simeq 1257 \frac{\text{Divisions}}{\text{Revolution}} \quad . \tag{8}$$

The task therefore was to find an optical shaft encoder capable of detecting bidirectional rotation with at least 1257 Divisions/Revolution in a package small enough to mount in the servo housing on the worm axis. There was no absolute size limitation; however, due to the construction of the servo it was desirable to find an encoder that was no more than 3 in. in diameter and no more than 1.5 in. in width. The companies that make shaft encoders are capable of custom building devices to meet a customer's specific needs. However, the prices are high, and the lead times are long for these special order parts. The encoder for this system needed to be reasonably priced and readily available to ensure timely completion of the project and to facilitate replacement, if necessary, in the future.

Optical shaft encoders are manufactured by numerous companies including Litton, Honeywell, BEI, IVO and Hewlett Packard. Sales literature from these companies was reviewed prior to making a decision on the specific shaft encoder model to be used. Incremental optical shaft encoders with resolutions that range from one pulse per revolution (PPR) to 2540 PPR are available off the shelf from one or more of these companies. While evaluating sources of supply, incremental optical shaft encoders were also found in use in various laboratories and shops in the Electrical Engineering and Physics Departments at the NPS. Encoders immediately available from stock included two Vernitech 1200 PPR (model VOE-23-1200-AI-LD5-2L1-1603-2) encoders from the Physics Department and two Hewlett Packard (HEDS-6000 J06) encoders from the Electrical Engineering Department. The use of the Vernitech encoders was ruled out because they were unable to detect bidirectional movement without increasing the complexity of the signal conditioning subsystem. Additionally, technical literature requested on two separate occasions from Vernitech was never received.

The HEDS-6000 J06 encoders have a resolution of 1024 PPR. Each encoder provides displacement information in the form of TTL logic level signals via two output channels. When the encoder is properly adjusted the two output signals have a 90° phase difference. This quadrature phase relationship permits these encoders to detect bidirectional displacements. Rotation in one direction will cause Channel A to lead

Channel B (in phase), while rotation in the other direction will cause Channel B to lead Channel A. [Ref. 9: p. 2]

The presence of two output channels in quadrature phase has an additional benefit that is useful in some applications. Since the amount of position information has essentially been doubled, if the signal conditioner is designed to detect both the leading and trailing edges of one of the output channels the resolution of the measurement system can be doubled. The difficulty with using this technique is that multiple oscillations about a single point cannot be detected as such. If the camera were to oscillate less than one half of a pulse width about a transition the signal conditioner would detect and erroneously count the multiple transitions. [Ref. 7: pp. 13-16]

With 1024 PPR, which is less than the 1257 PPR required to meet the 0.1 milliradian accuracy specification on the pan axis, the maximum resolution available, if the HEDS-6000 was attached to the worm, can be calculated as;

$$\frac{360^\circ}{(50 \times 1024)} = (7.03125 \times 10^{-3})^\circ \text{ Pulse}^{-1} \ . \tag{9}$$

Although this was not sufficient to satisfy the pan axis resolution specification of $(5.73 \times 10^{-3})^\circ$Pulse,$^{-1}$ all of the other performance criteria could be satisfied. The trade-off seemed reasonable and was approved prior to proceeding further with the system design.

E. THE DISPLAY

Selecting a method to display the final system output was certainly the least demanding task required in the design of the system. With an expected resolution on the pan axis of approximately 0.007° over a range of 360° (a ratio of about 1:51,500) an analog display seemed out of the question. A five or six digit digital display on the other hand offered a simple, reliable and cost efficient means of presenting the output. Constructing the displays from individual, seven-segment, common anode, LED devices, and the appropriate display drivers was a straight forward task.

F. THE SIGNAL CONDITIONER

Anticipating the use of two incremental shaft encoders as the transducers for the measurement system and the use of digital readouts as the display devices significantly reduced the number of possibilities for the signal conditioning subsystem. In addition to the hardware and software described in each of the subsequent discussions, each

technique listed here would require an edge detector to detect the transitions in the TTL signals from the shaft encoders.

- Up/down counter with table look-up.
- Up/down counter and multiplication.
- Add/subtract.
- Microcomputer.
- Microprocessor.

1. Up/down (U/D) counter with table look-up

This technique would involve the use of a shift register, some associated logic, an U/D counter and a table look-up device such as an erasable programmable read only memory (EPROM). The shift register would serve as a hardware buffer that could be used to account for the hysteresis introduced by the worm-wormgear assembly. The length of the buffer would have to be determined experimentally. The logic associated with the shift register would determine the "validity" of each transition (count) signal from the edge detector by checking the contents of the shift register and comparing the current direction of rotation with the previous direction of rotation. Whenever the logic detected a change in rotation direction, or a partially full, or a partially empty buffer (depending on the direction of rotation), the transition would not represent a "valid" count since any of these conditions would indicate that the transition was due to a change in the position of the worm without a corresponding change in the position of the wormgear. Thus, the present transition would be the result of hysteresis due to the backlash in the gear train and the transition would be "invalid". Such a transition would cause the contents of the hysteresis buffer to be modified appropriately. A more complete discussion of this topic is contained in Chapter IV. The shift register and associated logic could be built from common TTL devices readily available at the NPS.

Once a count had been determined to be valid, an U/D counter would be used to keep a running total of the number of counts. Again, such a device could be built using readily available TTL devices such as the 74LS168A, Synchronous 4-Bit Up/Down Decade Counter, or the 74LS169A, Synchronous 4-Bit Up/Down Binary Counter. The output from the U/D counter would then be used as an address to "look up" a predetermined number stored in an EPROM.

This design approach was considered relatively straightforward, and at least initially seemed like a viable option. Its major advantage was that it was conceptually quite simple. This simplicity had a price though; it would have been very hardware in-

tensive and consequently would have been a very inflexible design. For example, if at a later date the use of an arbitrary reference was desired, additional hardware would have to be added to the signal conditioner so that the counter could be initialized to the arbitrary starting point. Since all of the processing would have to be done in hardware, even a minor modification in the system could necessitate a major design revision. Another consideration was that saving in excess of 51,000 unique positions in the EPROM presented a nontrivial problem that would need to be solved if this technique was used.

2. Up/down counter and multiplication

This is very similar to the previous case, differing only in that instead of looking up a predetermined position, in a memory device, a multiplier would be used to multiply the output of the U/D counter by a predetermined constant. As an example, if experimental results indicated that the shaft encoders generated one pulse for every 0.00703125° that the camera was displaced, then each time a valid count was received, the updated count would need to be multiplied by 0.00703125 in the final stage of the signal processor. The result would be the new position. Again, while conceptually simple this idea had some significant disadvantages. Even more hardware intensive than the first approach, this design would also have extremely limited flexibility.

Accomplishing the multiplication would have posed a formidable task. The scale factor would have to be written as some integer (e.g., 0.00703125 would become 703125, or 70323, or 7031, etc.) depending on the desired accuracy. The count, already an integer, would be multiplied by the constant and the correct position of the decimal point would have to be determined. Locating the decimal point would be a relatively simple task, but multiplication of two numbers such as 51,000 and 70,313 would not be as easy. At least one 16 bit by 16 bit binary multiplier (i.e., TRW's MPY016H) is currently available, but since the number 70313 cannot be represented in binary by less than 17 bits, the multiplication could have been accomplished in either of two ways. One solution would have been to perform the multiplication in two or more stages. Alternatively, 70313 could be "rounded" to 7031. The second option would create a cumulative round-off of about $(1.28 \times 10^{-3})°$ per 360° rotation (assuming 51,200 pulses per 360°). This round-off error would probably have been acceptable, but the complexity and limited flexibility of either multiplication scheme made the U/D counter and multiplication an unacceptable candidate for the signal processing subsystem.

3. Add/subtract

A third design concept considered the elimination of the U/D counter altogether. The same logic proposed for use in the previous two designs could have been

used to check the validity of a count and to determine the direction of rotation. However, instead of valid counts being sent to an U/D counter as before, these counts would now signal an adder to add or subtract a predetermined constant from the running total. Just as before, this simple idea could probably have been made to work at least once, but its usefulness as a part of a larger system would certainly have been limited. The large number of components required to realize this design would have increased the probability of failure, complicated troubleshooting and reduced overall flexibility.

4. Microcomputer

If the signal processing subsystem was built around a microcomputer (PC), virtually all of the disadvantages associated with the previously discussed approaches would be eliminated. Since all of the logic could be implemented in software, modifications to the system would be relatively simple to make, and the system's flexibility would be enhanced. However, dedicating a microcomputer, even an inexpensive model, to the signal processing tasks for this measurement system was considered overkill, and timesharing with one of the PC's already in service was possible, but not considered practical or convenient since these microcomputers had already been dedicated to a variety of tasks.

5. Microprocessor

One final option for the signal processor remained. Microprocessors are relatively inexpensive and powerful and are available in a wide variety of makes and models. If a microprocessor was used instead of a microcomputer or a straight hardware processor, the "nice to have" requirements such as flexibility, ease of modification and capacity for expansion, as well as the required signal processing functions could all be satisfied. On the other hand, microprocessors have one distinct disadvantage; they are not user friendly. The use of a microprocessor implied countless hours spent tracking individual bits, debugging assembly language code, studying timing diagrams, etc.. No matter how distasteful the thought, however, a microprocessor was clearly the best way to perform the signal processing functions of the measurement system.

As with the selection of the transducer, once the basic design approach had been determined, selection of a specific device was a relatively straight forward task. The choice of one microprocessor over another was a function of the processor's ability to perform the required tasks, cost effectiveness and availability. Motorola's MC68705U3 seemed to satisfy all of these requirements. The MC68705U3 is a four kilobyte EPROM microprocessor, built using HMOS (high-density NMOS) technology with an eight bit architecture. The "68705" operates on a 5.0 volt dc supply, has 112 bytes of on chip RAM, four vectored interrupts, 24 TTL/CMOS compatible bidirectional I/O lines (eight

lines are LED compatible), eight dedicated input lines and an internal eight bit timer with a seven bit programmable prescaler. [Ref. 10: p. 1]

In Ref. 10 Motorola advertises the following software features:

- Programming language similar to the 6800 family.
- Byte efficient instruction set.
- Easy to program.
- True bit manipulation.
- Bit test and branch instructions.
- Versatile interrupt handling.
- Powerful indexed addressing for tables.
- Versatile index register.
- A full set of conditional branches.
- Memory usable as registers/flags.
- Single instruction memory examine/change capability.
- Ten powerful addressing modes.
- All addressing modes apply to EPROM, RAM and I/O.

One key advantage to using the MC68705U3 was that one of the microprocessors, and a Motorola M68705EVM (the evaluation/programming module for the M6805 family of devices) were both available for immediate use at the NPS. The fact that the 68705 utilized HMOS technology suggested that it should be a relatively low cost microprocessor. In fact, except for the price of a phone call, a second microprocessor was obtained free of charge from a local electronics wholesaler. Additional microprocessors, to be used as replacements and as backup devices were ordered for about $20.00 each.

One potential problem with using a microprocessor begged checking prior to proceeding. Initial measurements of the camera's maximum rotation velocity produced the following results.

$$\omega_{PAN(\max)} \simeq 1.0 \text{ rpm} \tag{10}$$

$$\omega_{TILT(\max)} \simeq 0.5 \text{ rpm} \tag{11}$$

Again the requirements for the pan axis presented the most stringent design limitations. Since the optical shaft encoders were anticipated to deliver 51,200 PPR (1024×50) the

microprocessor on the pan axis had to be capable of processing 51,200 pulses/minute (1.172 ms / pulse). The MC68705U3 is designed to operate with an oscillator frequency (f_{osc}) of between 0.4 MHz and 4.4 MHz and has an instruction cycle time ($4/f_{osc}$) of between 0.950 μs and 10 μs. Assuming that a 4.0 MHz clock was used, the instruction cycle time would be 1.0 μs. This would mean that the processing of each pulse would have to be accomplished in no more than 1172 instruction cycles to ensure that no pulses would be missed.

III. DESIGN

A. GENERAL

The schematic diagrams for the measurement system are shown in Appendix A; Appendix B contains copies of the printed circuit board plans. Before describing the detailed operation of each of the individual components of the measurement system, a brief overview of its basic theory of operation is in order. The measurement system is actually two separate systems operating independently. The system designed to measure the azimuth or pan angle will subsequently be referred to as the "Pan System", and the other system, designed to measure the elevation or tilt angle, will be referred to as the "Tilt System". However, because the two systems are quite similar the discussion which follows will only specifically describe the operation of both systems where their operation differs.

In the most general terms, the measurement system shown in Figure 2 on page 6 uses two microprocessors to count the pulses generated by the incremental shaft encoders, and to provide output signals to the display devices. The counting function performed by each microprocessor involves determining the direction of rotation and determining whether each count is "valid". Valid counts are encoder pulses which result from the displacement of the camera servo, while invalid counts are a result of the hysteresis in the gear train. The number of valid counts is directly proportional to the angular displacement of the camera axis.

Both of the displays are capable of presenting the position information in two basic forms. In the Count Mode they display the number of pulses that have been detected, and in the Position Mode they display the angle in degrees that the number of pulses represents. A third display mode, which is a combination of the first two, is also available. In the third mode, referred to as the "Blinking Mode", the display will alternately display the count and the angle.

B. SHAFT ENCODER AND LINE DRIVER

Figure 11 shows the basic components of the HEDS-6000 incremental optical shaft encoder. The encoder is approximately 56 mm in diameter and 20 mm deep. The code wheel assemblies for the HEDS-6000 J06 encoders are designed to mount on 0.25 in. shafts. Since the mounting surface shown in Figure 11 is not part of the encoder kit, and did not existed on either of the worm axes, two such surfaces were machined in the

Physics Department Shop and attached to one end of each of the worms. Each surface has a 0.25 in. diameter shaft to which one of the code wheel assemblies is attached. Two sheet metal brackets, also made in the Physics Department Shop, are bolted to the servo frame. The encoder bodies are mounted to these brackets. Figure 12 shows the shaft encoders mounted in the camera servo. [Ref. 9: p.1]

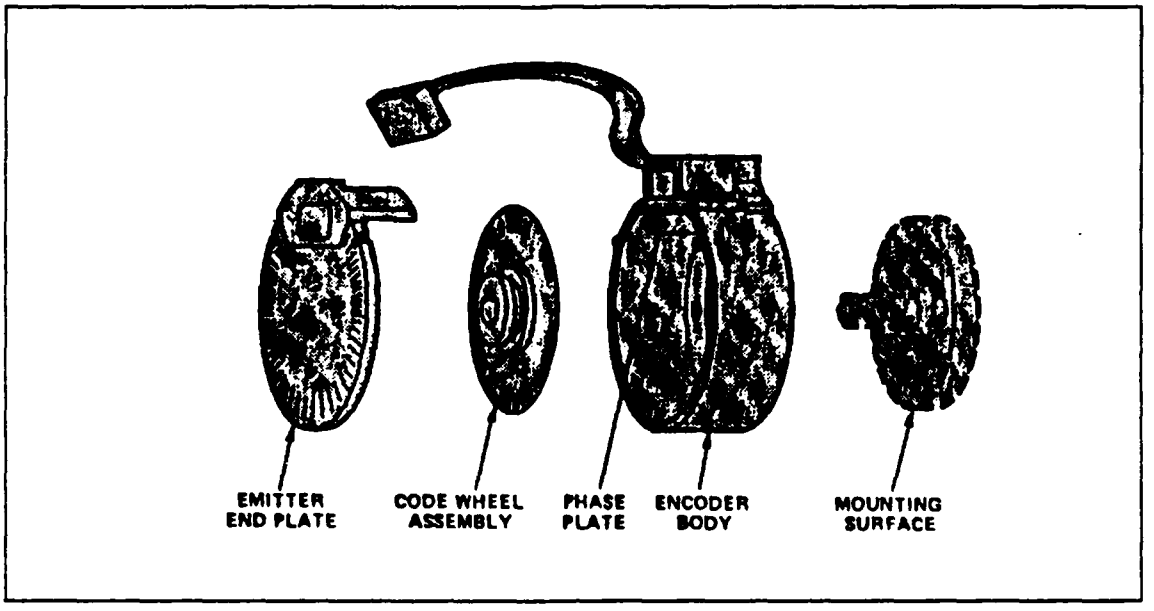


Figure 11. HEDS-6000 Series Encoder Kit: From Ref 9: p.6.

Also seen in Figure 12 is a printed circuit board mounted between the two optical shaft encoders. Each of the encoders is electrically connected to the board via a separate ten wire ribbon cable. The power and the output signals from the shaft encoders are transmitted through these cables. A sketch of the ribbon cable connector and the pinout for the connector are shown in Figure 13. Each connector is attached to a ten pin header on the printed circuit board. Pins 2, 7 and 9 are connected to a +5.0 Vdc power supply external to the camera servo. A description of the power supply is given later in this chapter. Pins 3, 4, 5 and 6 are connected to the power supply ground. The HEDS-6000 does not have an index pulse, therefore pin 10 is not connected.

The remaining two pins on each header connect the output channels of the shaft encoders to two 74S140 line drivers. Each 74S140 is a dual four-input NAND gate 50-Ohm line driver. The line drivers are powered by the same supply as the encoders. They serve as buffers between the encoders and the transmission lines which are used to transmit the encoder signals to the signal conditioner. The line drivers' typical high

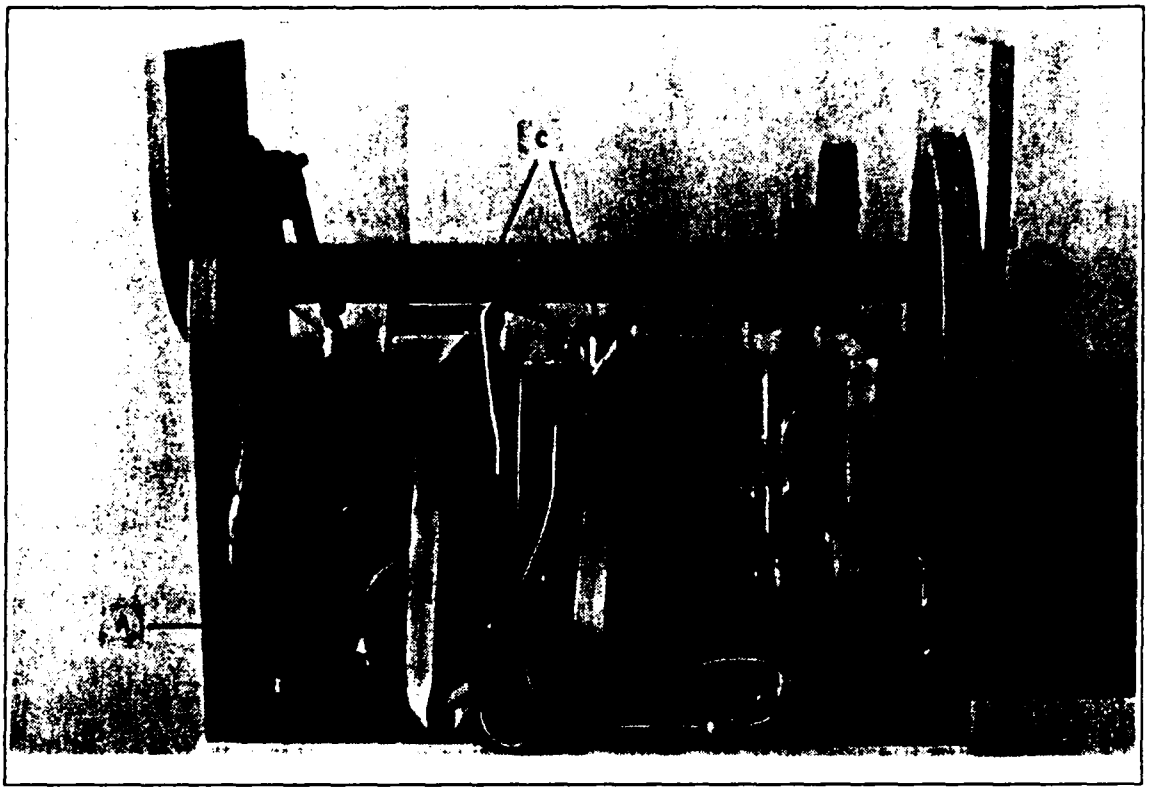


Figure 12. Modified Camera Servo: (a) Optical shaft encoder used to measure Pan axis displacement, (b) Optical shaft encoder used to measure Tilt axis displacement, (c) Ribbon cable, (d) Printed circuit board.

output current is 10 mA and the maximum is 18 mA. Resistance in the transmission line was measured to be approximately 35 Ω, and to be on the safe side a 10 Ω connector loss was assumed. The voltage drop due to an 18 mA current through a 45 Ω resistance is 0.81 V. A "voltage high" signal received at the signal conditioner should therefore be about 4.19 V. This is well above the maximum, positive-going threshold voltage specification of 2.0 V for the 7414 Schmitt triggers, which are used to receive the signals at the signal processor. [Ref. 11: pp. 5-73, 5-74, 6-44]

C. ENCODER-MICROPROCESSOR INTERFACE

The signals transmitted by the two line drivers are Channels A and B of each of the shaft encoders. These signals contain the raw data which the signal processor converts into position information. Three pairs of multiple pin connectors are used in the encoder-microprocessor interface. The pinouts for these connectors are shown in

BOTTOM VIEW

PINOUT

PIN #	FUNCTION
1	CHANNEL A
2	V_{CC}
3	GROUND
4	N.C. OR GROUND
5	N.C. OR GROUND
6	GROUND
7	V_{CC}
8	CHANNEL B
9	V_{CC}
10	CHANNEL I

Figure 13. Encoder Connector Specifications: From Ref. 9: p. 6.

Figure 14. Each of the four signals is received at the signal processor by a 7414 Schmitt trigger which is used to "clean up" the signal. The Schmitt triggers lower the system's susceptibility to errors caused by slow state transitions and increase the signals' fan out capabilities [Ref. 7: p. 13].

The output of each Schmitt trigger is routed to an input port of the appropriate microprocessor. Each Channel A signal is also the input to an edge detector. The edge detectors each consist of three 7414 inverters, a 74LS86 EXOR gate and a 47 nF capacitor, configured in the manner shown in Figure 15. This configuration causes the interrupt line to go low for approximately 2 μs each time Channel A transitions from low to high or from high to low. Since an oscillator frequency, f_{osc}, of 4.0 MHz is being used, the interrupt pulse width, t_{WL}, must be greater than or equal to 1.25 μs [Ref. 10: p. 3]. The value of the capacitor required to achieve the 2 μs delay was determined experimentally.

D. SWITCHES

The measurement system has 10 switches that allow the operator to control specific functions of the signal processors and the displays. Figure 16 shows the physical location of these switches on the control panel. Switches SW1(P) and SW1(T) control the reset lines to the microprocessors. Switches SW2(P), SW2(T) and SW5 control the display mode. The Function and Set switches; (SW3(P), SW3(T), SW4(P) and SW4(T)) allow the user to change the length of the hysteresis buffer in the microprocessor. The

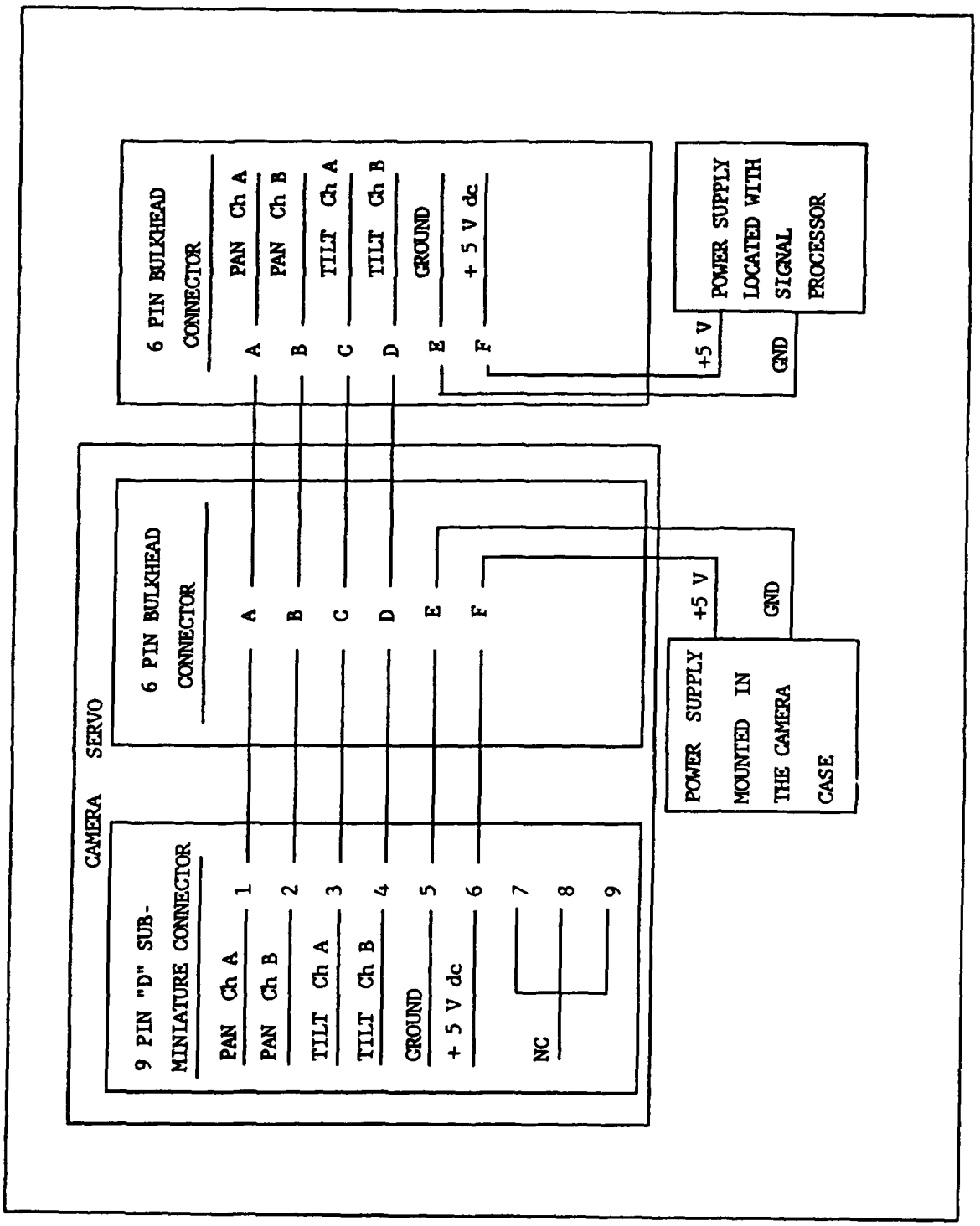


Figure 14. Encoder-Microprocessor Connector Specifications

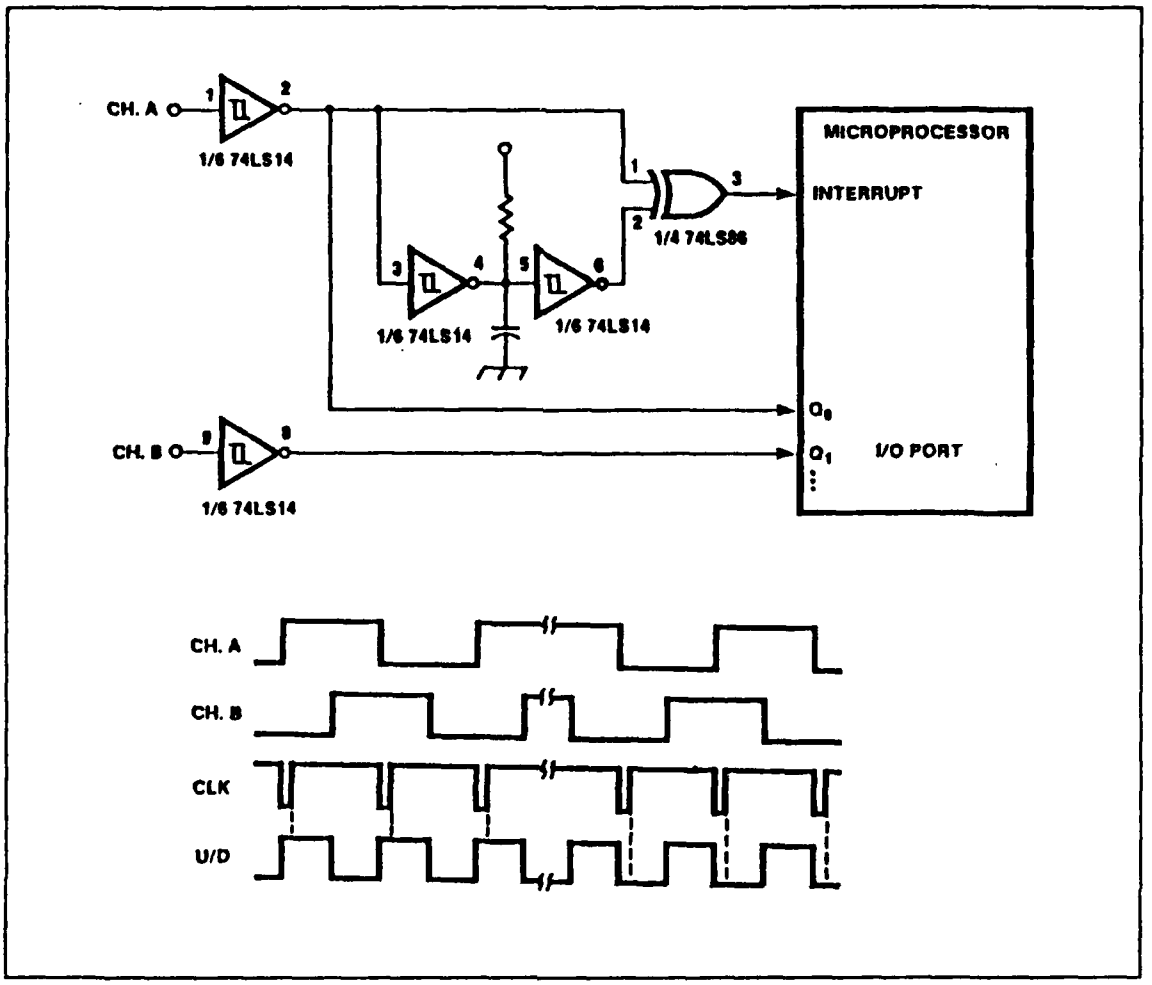


Figure 15. Interrupt Interface: After Ref 7: p. 14.

master power switch is SW6. Table 2 identifies the switches by name, description and function.

Any mechanical switch will "bounce" or "chatter" when it is thrown, and since the operation of the signal processor depends on the number of times that the Display Mode and Set switches change position, these switches had to be "debounced". A very simple but effective way to do this is with an $\overline{RS}$ latch. Switches SW2(P), SW2(T), SW4(P) and SW4(T) are each debounced in this manner. Each latch is made from two 74S00 NAND gates connected in the manner shown in Figure 17. [Refs. 13: pp. 132-135, 12 : pp. 3,4]

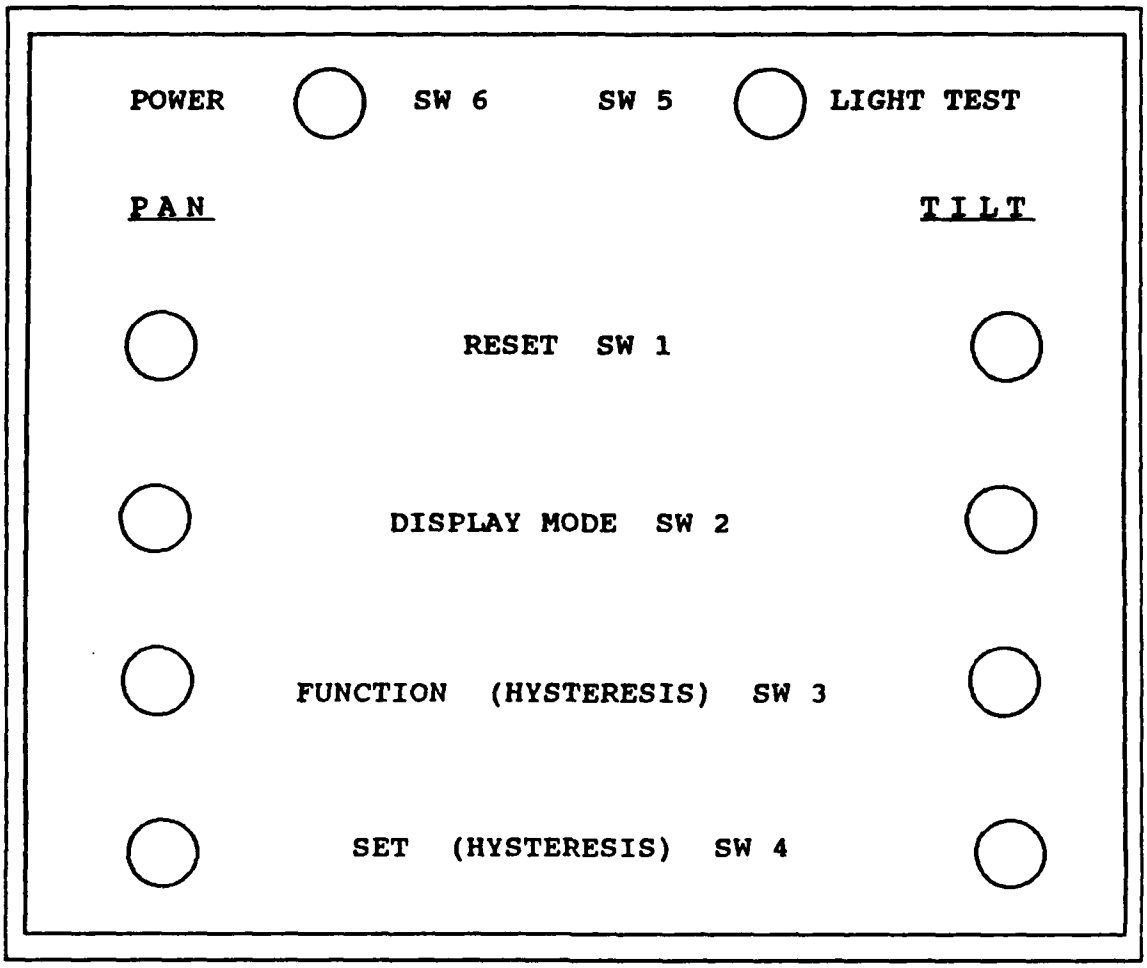


Figure 16. Control Panel

E. MC68705U3

1. General

The two 40 pin MC68705U3 microprocessors (MPU's) are the heart of the measurement system. With the exception of the light test signal, every signal in the system is either part of the input to one of the MPU's or part of their output. The MPU's were programmed using the assembly language syntax, assembler directives and instruction set for the M6805 family of microprocessors which are described in Ref. 14. The Pan and Tilt programs are listed and their operation is described in Appendix D. The pin assignments for the MC68705U3 are shown in Figure 18. Table 3 briefly describes the purpose of each pin and the actual connections for the two MPU's.

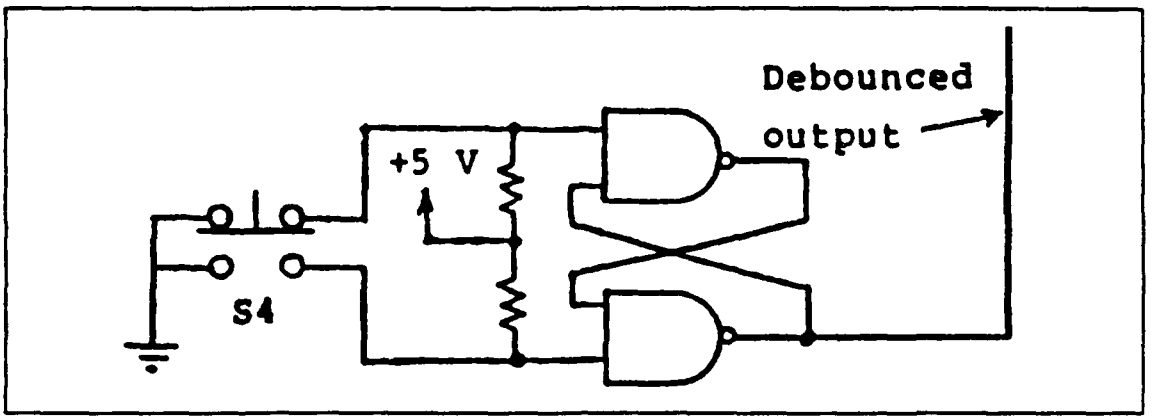


Figure 17. Debouncing circuit: After Ref 12: p. 4.

2. Memory Map

a. Input/Output (I/O)

The memory map for the MC68705U3 is shown in Figure 19. The digits following a "S" are the hexadecimal representation of the address for a specific memory location. The data registers occupy the first four memory locations of each MPU. Thus the information written into the registers at S000, S001 and S002 is written to the output ports A, B and C respectively. Port D, at address S003, is an input only port as indicated in Table 3. In order to determine the state of the input lines, the contents of the register at S003 must be read by the MPU. Registers S004, S005 and S006 are the data direction registers (DDR's) for Ports A, B and C respectively. Because all three ports are used as "output only" ports, in this application, the DDR's are all established as such by an initialization routine performed by the MPU's during their initial power-up and after each external reset. [Ref. 10: pp. 5, 12, 14]

Pin 18 of the MPU can be used as either a general purpose input line or as an interrupt line. The primary interrupt line on each MPU is used to signal the occurrence of a state transition on Channel A. Pin 18 is used as a second interrupt line to signal a display mode change request from the operator. The Miscellaneous Register (MR) at address S0A is used to control the operation of the second interrupt line ($\overline{INT2}$). In order to establish pin 18 as an interrupt line, bit 6 of the MR is cleared by the Initialization Routine. The $\overline{INT2}$ Interrupt Request Bit, bit 7 of the MR, is cleared by default upon reset. It is set when a falling edge is detected on the Display Mode line which is connected to pin 18. When this occurs and bit 6 of the MR is cleared, an interrupt request is generated. This interrupt request causes the display mode to change.

Table 2. SIGNAL PROCESSOR AND DISPLAY SWITCHES

NAME	DESCRIPTION AND FUNCTION
PAN RESET (SW1(P))	Momentary action push button switch: Resets the Pan Microprocessor. Causes the Pan Display to be reset to zero.
TILT RESET (SW1(T))	Same as SW1(P) except it affects the Tilt System only.
PAN DISPLAY MODE (SW2(P))	Single pole double throw toggle switch: Each time the switch position is changed the Pan Display toggles from Count mode, to Position Mode, to Blinking Mode, to Count Mode, etc..
TILT DISPLAY MODE (SW2(T))	Same as SW2(P) except it affects the Tilt Display only.
FUNCTION (HYSTERESIS) (SW3(P))	Single pole double throw switch: When closed, causes the "Function" line of the Pan Microprocessor to go low and causes the size of the hysteresis buffer to be displayed on the Pan Display. Enables SW4(P).
FUNCTION (HYSTERESIS) (SW3(T))	Same as SW3(P) except it affects the Tilt System only.
SET HYSTERESIS (SW4(P))	Single pole double throw switch: Inoperable unless SW3(P) is closed. If SW3(P) is closed, each time the position of SW4(P) is changed the length of the hysteresis buffer is incremented by one. Maximum buffer length is 25. Toggling SW4(P) when the buffer length is 25 will cause the buffer length to be reset to zero.
SET HYSTERESIS (SW4(T))	Same as SW4(P) except it affects the Tilt Hysteresis buffer.
LIGHT TEST (SW5)	Single pole single throw switch: When closed, lines 1, 2, 7, 8, 10, 11 and 13 on each of the LED displays will go low. Unless an element is burned out, every digit in both displays should be an eight.
POWER (SW6)	Single pole single throw switch: When closed, applies + 5.0 V dc power to the signal processors and the displays.

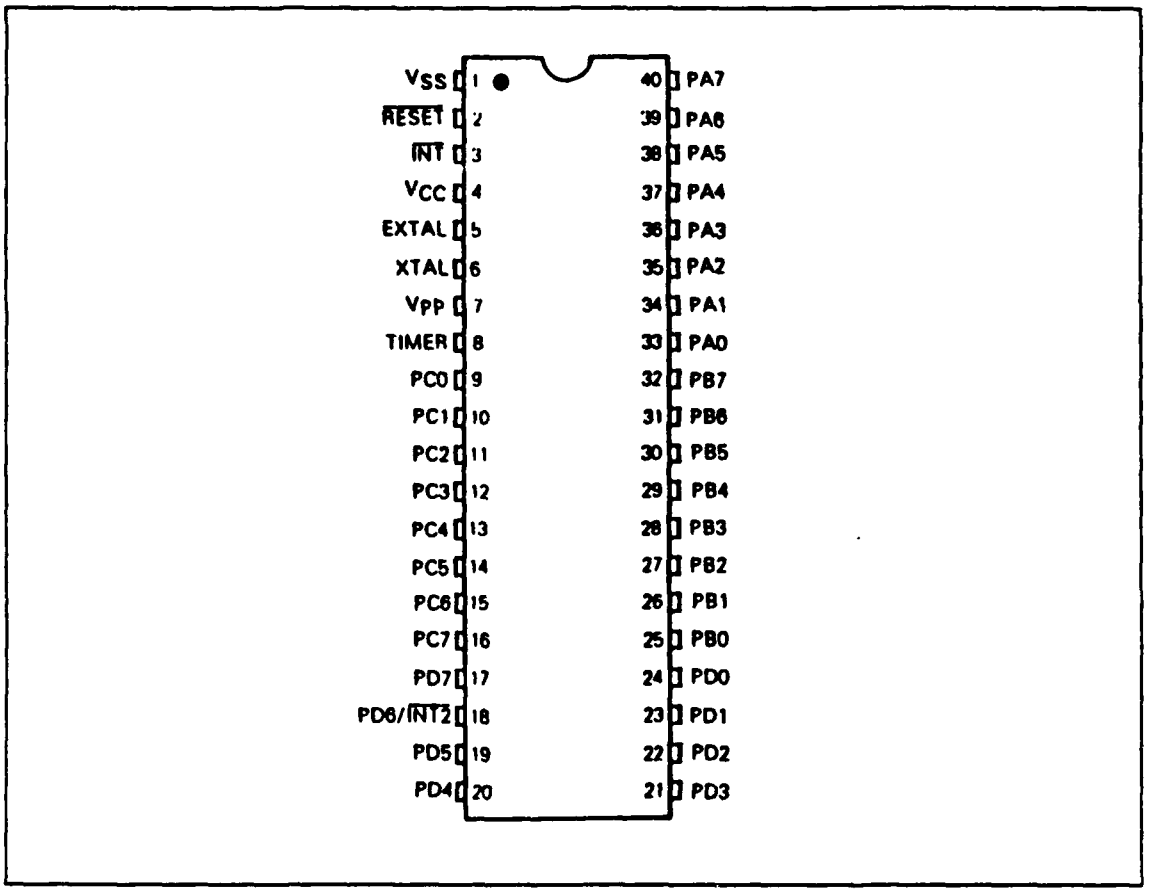


Figure 18. MC68705U3 Pin assignments: From Ref. 10: p. 1.

Once bit 7 has been set by an interrupt, it must be cleared by software to avoid repeated and unwanted interrupts from occurring. This task is performed by the Mode Change Routine in each EPROM. [Ref. 10: pp. 1, 10, 13, 15]

b. Timer

The operator can cause either or both of the displays to "blink" by using the Display Mode Switches, SW2(P) and SW2(T). When one of the systems has its display in the Blinking Mode, the associated MPU uses its timer to generate a timer interrupt request every second. The interrupt request causes the MPU to execute the Mode Change Routine. A block diagram of the timer is shown in Figure 20. The timer consists of an eight-bit counter which is decremented toward zero by the f_{CIN} input. When the counter reaches zero, it sets the Timer Interrupt Request Bit (TIR) of the Timer Control Register (TCR), and a timer interrupt request is generated unless the

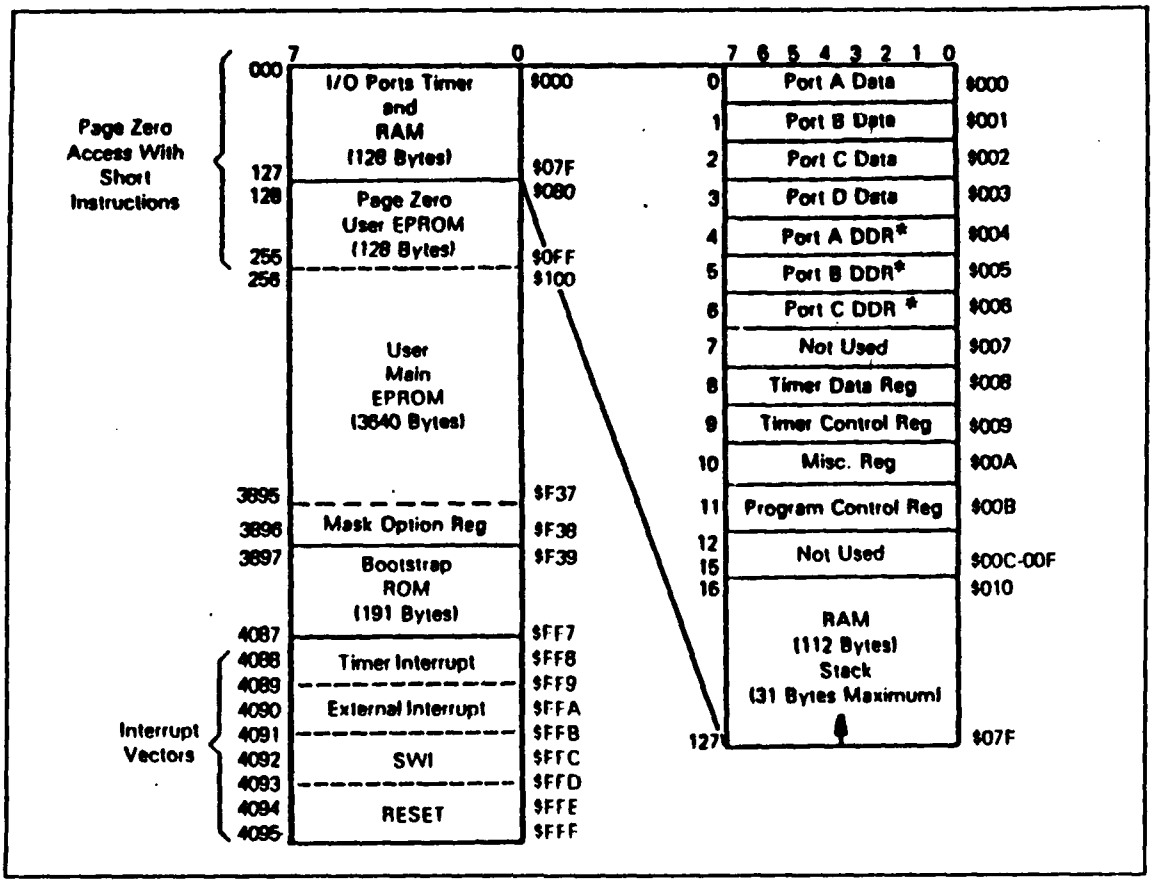


Figure 19. MC68705U3 Memory Configuration: From Ref. 10: p. 5.

Timer Interrupt Mask Bit (TIM) of the TCR is set. A brief description of each of the timer registers and their configuration follows.

(1) Timer Data Register (TDR). The TDR is the eight-bit counter which sets the TIR bit of the TCR when it decrements to zero.

(2) Timer Control Register. The contents of the TCR determine the general operation of the timer.

- Bit 7, Timer Interrupt Request (TIR), signals a TDR underflow when it is set and will cause a timer interrupt request if the TIM bit of the TCR is clear. The TIR is cleared by the MPU reset or by program control.
- Bit 6, Timer Interrupt Mask (TIM), inhibits a timer interrupt request when it is set. It is set by external reset or program control to inhibit the Blinking Mode, and is cleared by software when the Blinking Mode is requested by the operator.
- Bit 5, External or Internal Clock Select (TIN), is used to select the timer clock source. Since the internal clock is used in this application, the TIN bits of both

Table 3. MPU CONNECTIONS

Pin	Name	Description
1	V_{SS}	Ground
2	$\overline{\text{RESET}}$	When $\overline{\text{RESET}}$ is pulled low program execution halts, all variables are reinitialized and the Pan display is set to zero. SW1(P) controls the $\overline{\text{RESET}}$ line on the Pan MPU.
3	$\overline{\text{INT}}$	Allows asynchronous interruption of the processor. When $\overline{\text{INT}}$ is pulled low by the Count Edge Detector the MPU executes the "Count Routine".
4	V_{CC}	+5 V dc power connection.
5	EXTAL	External clock input. Connected to a 4.0 MHz external clock which provides the MPU system clock.
6	XTAL	Crystal clock input. Connected to ground since an external clock is used.
7	V_{PP}	Programming voltage pin. Connected to V_{CC} for normal operation.
8	Timer	External timer control input. Connected to V_{CC} since the internal timer is used.
	Port C	General Purpose I/O lines.
9	PC0	The two least significant digits in the display are represented in binary coded decimal (BCD) by these eight lines.
10	PC1	
11	PC2	
12	PC3	
13	PC4	
14	PC5	
15	PC6	
16	PC7	
	Port D	General Purpose input lines.
17	PD7	PD7 is the Channel A input to the MPU.
18	$\frac{\text{PD6}}{\overline{\text{INT2}}}$	PD6 is used as a second interrupt line. When PD6 goes low the MPU changes display modes.
19	PD5	PD5 is the Channel B input into the MPU.
20	PD4	PD4 is the Function input into the MPU.
21	PD3	PD3 is the Set input into the MPU.
22	PD2	PD2-PD0 are not used and are tied to ground.
23	PD1	
24	PD0	

Table 4. MPU CONNECTIONS (CONT'D.)

Pin	Name	Description
	Port B	General Purpose I/O lines (LED compatible).
25	PB0	The most significant digit is represented in BCD by PB0-PB3 except on the Tilt MPU where these lines are connected to ground. PB4 determines which digits in the display are blanked. PB5 is not used. Connected to ground. PB6 determines the presence or absence of the display minus sign. PB7 determines the presence or absence of the display decimal point.
26	PB1	
27	PB2	
28	PB3	
29	PB4	
30	PB5	
31	PB6	
32	PB7	
	Port A	General Purpose I/O lines.
33	PA0	The third and fourth least significant digits are represented in BCD by these eight lines.
34	PA1	
35	PA2	
36	PA3	
37	PA4	
38	PA5	
39	PA6	
40	PA7	

MPU's are always cleared. For the same reason the Timer pins are connected to V_{CC} (see Table 3).

- Bit 4, External Enable (TEE), is not used by the measurement system. By keeping the TEE clear at all times, the 68705's internal timer is used exclusively.
- Bit 3, Prescaler Clear (PSC), is not used in this application; always cleared.
- Bits 2-0, Prescaler Select (PS2, PS1 and PS0), are always set during program execution. This causes the internal timer signal frequency to be divided by 128.

(3) Mask Option Register (MOR). Unlike the TDR and TCR the MOR is not software programmable; instead, it is implemented in EPROM.

- Bit 7, the Clock bit, is cleared to allow operation of the external 4.0 MHz clock.
- The Timer Option bit (TOPT), bit 6, is also cleared in this application. This permits the TCR to be software programmable.
- Bit 5 is cleared to permit the use of the internal clock with the timer.
- Bits 4 and 3 are not used.

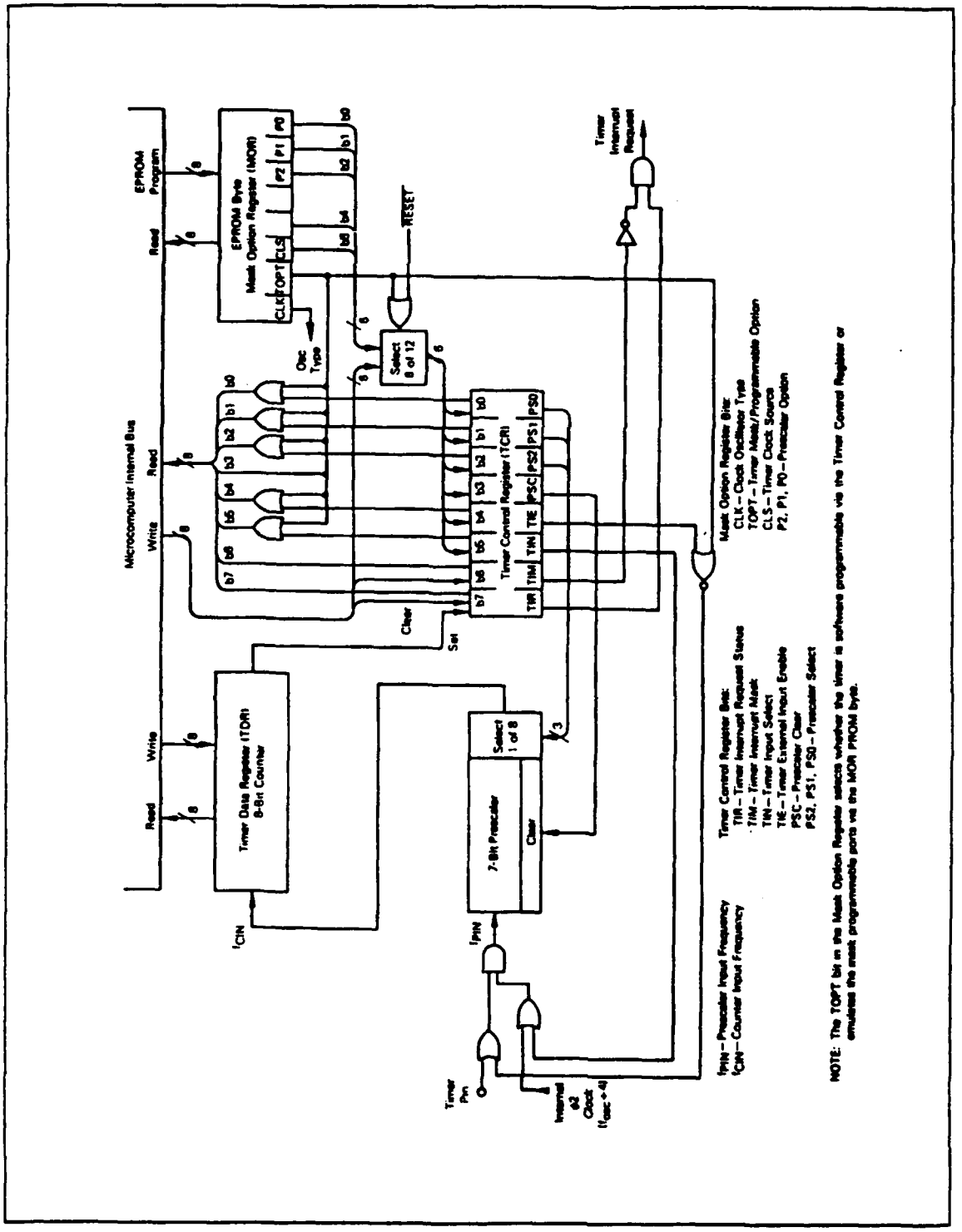


Figure 20. MC68705U3 Timer Functional Block Diagram.: From Ref. 10: p. 8.

- Bits 2-0 are all set and serve the same function as PS2, PS1 and PS0 bits of the TCR. [Ref. 10: pp. 6-8, 13-15]

3. RAM

The MC68705U3 has 112 bytes of RAM. The 112 bytes includes 31 bytes that can be used for the stack. Use of the stack is quite limited. During interrupts it is used to save the contents of the CPU registers and the program counter. During subroutine calls only the program counter is saved. The user's program has no other access to the stack. The programs written for each of the MPU's require less than 25% of the available RAM. The Pan MPU uses 27 of the 112 available bytes and the Tilt MPU uses only 22 bytes. The programs listed in Appendix D explain the function and give the location in memory for each of the variables. [Ref 10: p. 5]

4. ROM

The 3776 bytes of user EPROM in the MC68705U3 are divided into three separate blocks in the memory. Page Zero User EPROM is the ROM located between address S080 and S0FF. Because these addresses are only one byte long, instructions located in Page Zero ROM can be referenced with addressing modes not permitted with instructions located in other parts of the memory. Between address S100 and SF37 is the User Main EPROM. This portion of the memory in each MPU contains the vast majority of the signal conditioning programs. Located in another portion of the EPROM are the Interrupt Vectors. In each of these locations is the address of the first instruction the MPU is to execute when a particular interrupt occurs.

As with the RAM only a fraction of the available EPROM has been used in this application. The Pan MPU uses 899 bytes of the 3776 available and the Tilt MPU uses only 767 bytes. Since both programs are so small, one could reasonably ask why the two programs were not both put in one MPU. The primary problem with this idea is that each microprocessor can perform only one operation at a time. As indicated at the end of Chapter II, if the camera servo is rotating about its vertical axis at its maximum velocity of 1 rpm, the Pan signal conditioner must be capable of counting and displaying 51,200 counts per minute. Using a 4.0 MHz clock this allows the MPU 1172μs to count each pulse. Similarly the Tilt MPU has 1758μs to count each pulse when the camera is rotating at its maximum velocity about the horizontal axis. Assuming that the camera is rotating at its maximum velocity on both axes at the same time, and one MPU is being used to count the pulses from both encoders, the MPU needs to count 76,800 pulses per minute, which only allows 781.3 μs per pulse. The Pan MPU currently requires a max-

imum of 1032 instruction cycles or 1032μs to count a single pulse, and the maximum execution time for a single pulse on the Tilt axis is 825μs. Thus, a single 68705 lacks the computational speed required to ensure that no counts would be missed if it was used to process the data from both encoders.

5. Central Processing Unit (CPU)

> The CPU of the M6805 Family is implemented independently from the I/O or memory configuration. Consequently it can be treated as an independent central processor communicating with I/O and memory via internal address, data and control buses. [Ref 10: p. 6]

The CPU has five registers that are available for use by the operator. The function of each of these is described below.

- The Accumulator (A) is a general purpose data register used for arithmetic calculation and data manipulation.
- The Index Register (X) can be used as a second accumulator but is generally used for the indexed addressing mode. In the indexed addressing mode an effective address is created by adding the contents of X to a number provided by the instruction.
- The Program Counter (PC) contains the memory address of the next instruction to be executed by the MPU.
- The five bits of the Condition Code Register (CCR) keep information concerning the results of the last instruction executed by the MPU. Reference 14 gives a detailed description of each of the instructions in the M6805 Family Instruction Set and explains the effect of each instruction on the CCR. A brief description of each bit in the CCR follows.
  - The Carry (C) bit is set if a carry or a borrow was generated by the last arithmetic instruction. The state of the C bit can be software controlled.
  - The Zero (Z) bit is set if the result of the last arithmetic, logic or data manipulation instruction was zero.
  - The Negative (N) bit is set if bit seven of the result of the last arithmetic, logic, or data manipulation instruction is set.
  - The Half Carry (H) bit is set if an ADD or an ADC instruction causes a carry to occur between bits 3 and 4 of the result.
  - The Interrupt Mask (I) bit is set when an external interrupt ($\overline{\text{INT}}$) occurs. If another interrupt occurs (e.g. Timer Interrupt or $\overline{\text{INT2}}$) when the I bit is set, the second interrupt is latched so that it can be processed as soon as the I bit is cleared. The I bit can be set or cleared by software.
- The contents of the Stack Pointer (SP) are the address of the next available location on the stack. As previously discussed, the stack is only used to keep track of the PC during subroutine branches, and all of the CPU registers during an interrupt. [Refs. 10: p. 6, 14 : pp. 14-15]

6. Input

Each MPU uses six input lines. Two of these lines, $\overline{INT}$ and $\overline{INT2}$, are interrupt lines that detect a negative-going edge on their respective lines. The other four lines are general purpose input lines on Port D. All of the pins on Port D are TTL compatible which made the hardware design relatively straightforward. The electrical characteristics for the input pins are listed on p. 2 of Ref. 10.

The general operation of the two interrupt lines is described in Table 3 on page 39. Once they are understood, interrupts are a simple yet powerful tool. Only three of the four interrupts available on the MC68705U3 are used by the Pan and Tilt programs. The software interrupt is not used. When the MPU is interrupted, current program execution is halted, the contents of the CPU registers are placed on the stack, and the MPU fetches the contents of the appropriate interrupt vector from memory. After the interrupt vector has been fetched, the PC is moved to that address and execution of the interrupt routine begins. There is no ambiguity when an external interrupt occurs since there is a dedicated interrupt vector in memory. The timer interrupt and $\overline{INT}$ however share the Timer Interrupt Vector. When one of these interrupts occurs, the interrupt routine must determine the source of the interrupt by checking the TIR bit of the TCR and bit 7 of the MR to determine the source of the interrupt [Ref. 10: p. 11]. Normal program execution resumes at the point at which the interrupt occurred when the interrupt routine executes a return from interrupt (RTI) instruction.

The Function and Set lines on pins 20 and 21 are connected to the Function and Set switches. The operation of these switches is described in Table 2 on page 36. The remaining two input lines to each MPU are Channels A and B from the respective shaft encoders. The MPU programs use the information from these two inputs to determine the direction of rotation and to identify repeated oscillations about a single transition.

7. Output

Each MPU is designed to provide position information at its output in two basic forms. On the Pan axis, in the Count Mode a number between -51,200 and +51,200 constitutes the output while in the Position Mode the output is an angle between 0° and 360°. A five digit display with a minus sign is sufficient for the count display. Using a five digit display with a decimal point in the Position Mode permits the angle to be displayed to the nearest hundredth of a degree. This resolution is not quite as good as the resolution of the shaft encoders (±0.007°); however, final testing of the measurement system revealed that resolution is actually limited to about ± 0.02° on the Pan axis and

about $\pm 0.14°$ on the Tilt axis. The five digit display is therefore completely adequate for this system.

The Pan MPU uses 23 of its 24 output pins to represent the five digits, a minus sign and a decimal point. Each of the five digits is available in BCD form on four output pins of the MPU. The five digits are referred to as Digit 1, Digit 2, etc., with Digit 1 being the least significant digit and Digit 5 being the most significant digit. Port A has as its output the BCD representation of Digit 3 and Digit 4. Digit 1 and Digit 2 are represented by the output of Port C. The low four bits of Port B contain the BCD representation of Digit 5. These 20 output lines are the input to five 74LS47, BCD/7-Segment Decoder/Drivers, which decode the BCD signals and drive the common anode LED indicators. The output from Pin 29, PB4, is one input to a 74LS32 OR gate, the output of which is used to blank leading zeros out of the display. Pin 30 is not used and is tied to ground. The remaining two output pins drive two LED segments in the display. The signal on pin 31 turns the minus sign off and on, and the signal on pin 32 determines whether the decimal point is displayed.

The electrical characteristics of the I/O Ports are given on p. 4 of Ref. 10. The output characteristics of Ports A, B and C are compatible with the input characteristics of the 74LS47 and the 74LS32 given on pp. 4-59 and 4-48 of Ref. 11. The pins on Port B are capable of sinking 10 mA when Port B is configured as an output port. A 220 Ω resistor placed in series with each of the display segments limits the current to approximately 9 mA and permits the MPU to drive the decimal point and minus sign directly.

One consideration in the design of this system was to provide a system capable of being readily expanded to meet changing needs. To this end, in addition to being connected to the LED display devices via the 74LS47's, the BCD data lines are also connected to a header on each of the MPU circuit boards. If, at a later date, the position information needs to be used in another system, a jumper connected to each of the headers could provide the information with little or no modification.

The Pan and Tilt signal processing subsystems are virtually identical in the hardware used to implement them. The only difference is that the Pan system has a five digit display, and the Tilt system needs only four digits to display its position information. Consequently, the Tilt system does not use Digit 5, and pins 25-29 are tied to ground.

F. DISPLAY

Each digit represented in BCD at the output of the MPU is decoded by a 74LS47 BCD/7-Segment Decoder/Driver. The decoding devices each convert a four bit BCD representation of a number into seven signals that each drive a separate segment of a common anode, seven segment, LED display. The 74LS47 is capable of sinking 24 mA from each of the LED segments. Without a current limiting resistor between each of the output pins on the 7447 and the corresponding pin on the display element however, this maximum current is exceeded. When this happens the LED's have a very short life, the 7447 overheats and the system fails to function properly. The addition of a 220 Ω resistor in each branch limits the current to about 9 mA per segment, and permits trouble-free operation.

Three of the output pins on each MPU are not used as inputs to the 7447's. As discussed in the previous section, the Decimal and Minus lines each drive individual LED segments directly. The third line, also mentioned briefly in the preceding section, is used with the Blanking In/Ribbon Blanking Out (BI/RBO) signal from Digit 4's 7447 to determine the Ribbon Blanking In (RBI) signal into the 7447 which drives the display for Digit 3.

The term "blanking" simply means removing the leading zeros from the display. The two display modes available from the MPU's have different blanking requirements. In the Position Mode the three least significant digits are not blanked, while in the Count Mode, all but the least significant digit are blanked. The RBI and BI/RBO pins on the 7447's, the Blank line out of each MPU and the OR gates, connected as shown in the schematics in Appendix A, provide this capability.

The Light Test (LT) pin on each 7447 is connected to SW5. When the switch is closed, the Light Test line goes low and each of the output lines on each of the 7447's also goes low, thus sinking current from all of the LED segments simultaneously. This feature allows the operator to check for inoperable display segments.

G. POWER SUPPLIES

The two power supplies shown in Figure 2 on page 6 are each +5.0 Vdc supplies. The power supply which provides power to the shaft encoders and the line drivers is physically mounted in the camera housing. It was built by modifying the +12 Vdc auto iris power supply. This was accomplished using an LM7805 Voltage Regulator in the manner shown in Figure 8 of Appendix A. The auto iris requires only 100 mA at +12 Vdc for correct operation, and the LM7812 Voltage Regulator has an available

output current of 1.0 A [Refs. 15, 16]. The remaining 900 mA is available to the LM7805 to power the encoders and the line drivers. Reference 9 lists the maximum power requirement for the HEDS-600 as 40 mA at +5 Vdc, and Ref. 11 specifies the maximum power requirement for a 74S140 is 1 mA at +5 Vdc. Thus, the 84 mA requirement for the two encoders and four line drivers is well within the capabilities of the modified power supply.

The second power supply is capable of providing 6.0 A at +5 Vdc which is more than adequate to provide the 1.4 A needed by the signal processors and the display devices. The power supply also has 12 Vdc and -5 Vdc ports. To preclude the potentially disastrous results which might occur if the power supply were incorrectly connected to the signal processor/display devices, the circuitry shown in Figure 21 was included on each MPU and display printed circuit board.

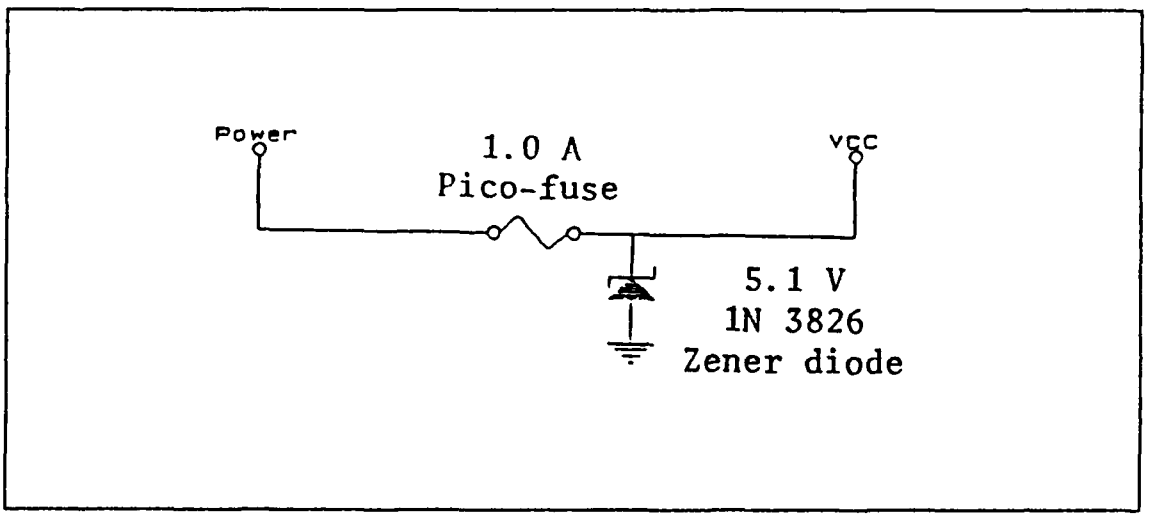


Figure 21. Reverse/Over-voltage Protection Circuit

IV. CALIBRATION , TESTING AND IMPLEMENTATION

A. GENERAL

Once the basic system design had been completed, and the MPU programs had been written, a prototype system was constructed. The prototype system might also be called the development model, since it was not only used to test the design , but was also used to calibrate the MPU programs. A block diagram of the prototype system is shown in Figure 22. The M68705EVM Evaluation Module (subsequently referred to as the EVM) provided the capability to debug and evaluate the MC68705U3-based signal processing subsystem. Operation of the signal processing MPU was performed by an MC68705U3 resident on the EVM.

The prototype system provided considerable flexibility in the testing and calibration of the system. The assembly language programs for the MPU's were written and edited on the PC. They were then assembled and linked using the 2500 A.D. 6805 Cross Assembler and 2500 A.D. Linker [Ref. 17: pp. (1-1)-(2-38)]. The result, a Motorola S19 output file (see [Ref. 17: pp. (A-1)-(A-4)]), was then down-loaded to the EVM using the file transfer program, Kermit. Downloading procedures are detailed in [Ref. 18: pp. (3-10)-(3-25),(3-37)]. The PC-EVM interface is shown in Figure 23.

After the program had been down-loaded into the MC68705U3 resident on the EVM, data entry and program debugging were controlled via the CRT monitor keyboard. The CRT-EVM interconnection is shown in Figure 24 and the monitor commands are described in [Ref. 18: pp. (3-8)-(3-25)].

The remainder of the signal processing functions were realized using hardware external to the EVM. These functions included edge detection of the output signals from the shaft encoders, decoding the output of the MPU, and generating the signals to drive the display devices. This portion of the prototype, referred to by [Ref. 18] as the "target system", was built on breadboards and is represented by the block in the center of Figure 22. The target system was connected to the MCU via a 40-pin jumper header, J1, on the EVM. The pinout for J1 is shown in Figure 25. The labels in Figure 25 refer to the labels used in the schematic diagrams which are shown in Appendix A.

B. CALIBRATION

Once the MPU programs were capable of counting the pulses generated by the shaft encoders, the programs needed to be "calibrated". This calibration procedure required

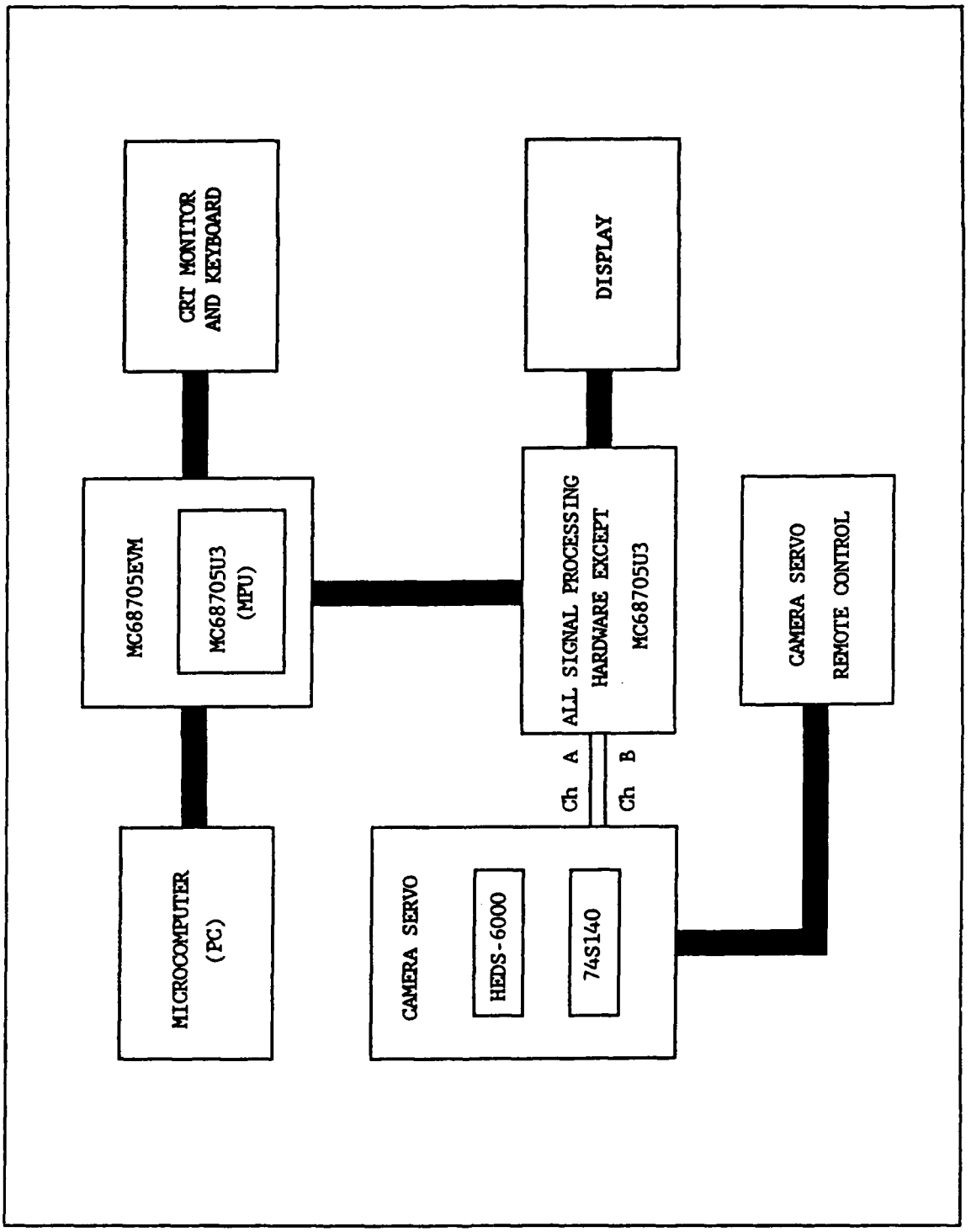


Figure 22. Prototype/Development Model

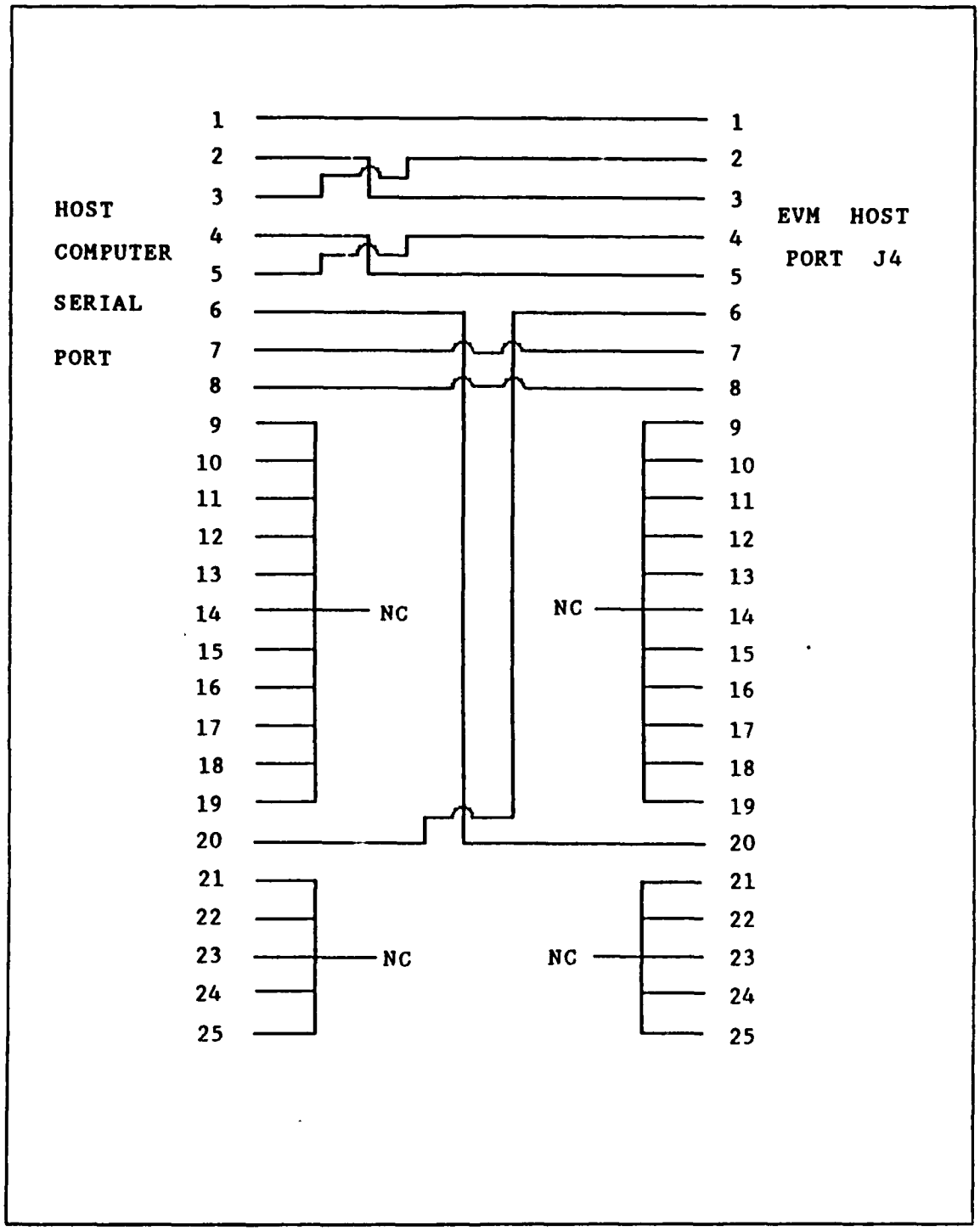


Figure 23. Host Computer - Evaluation Module Connections

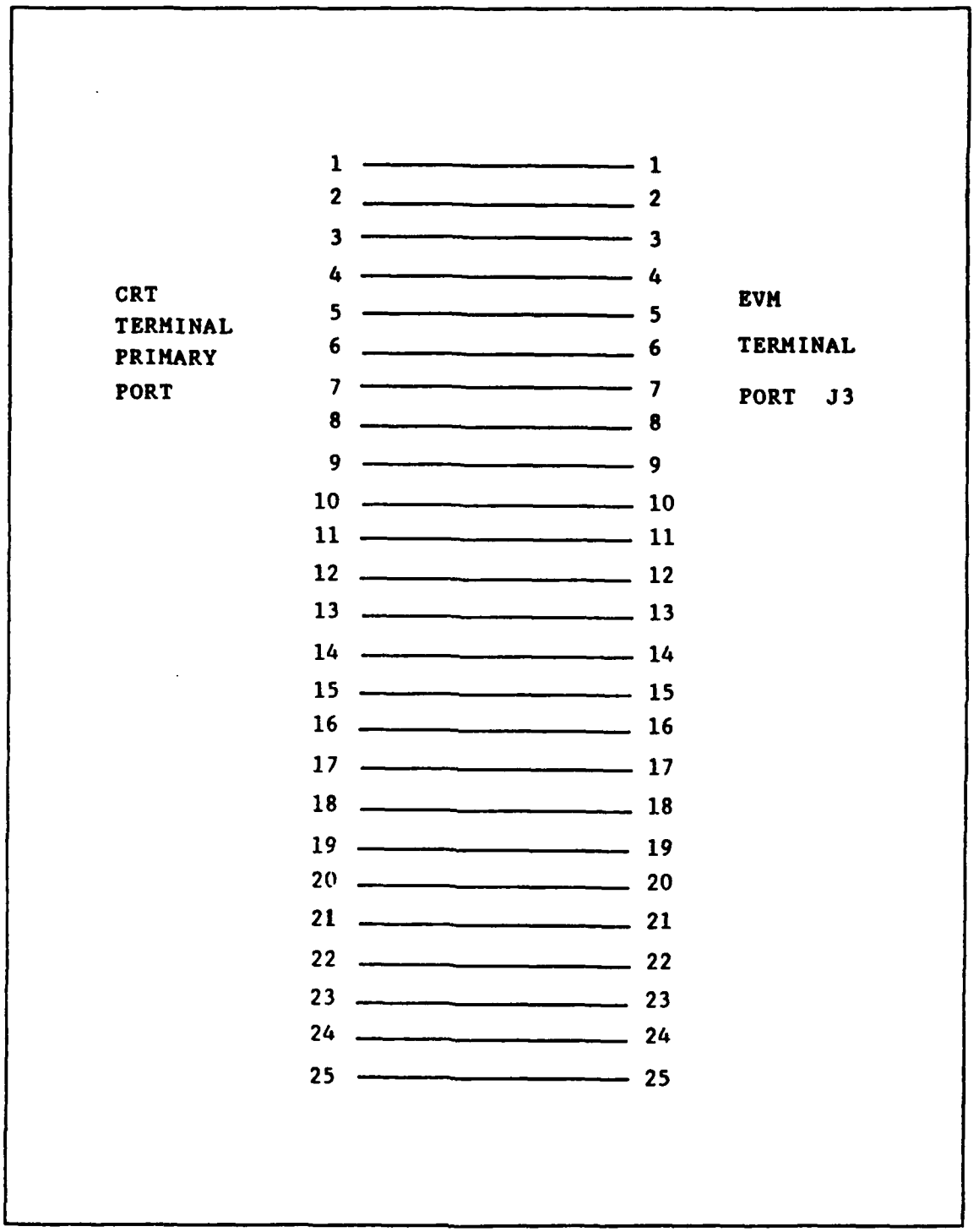


Figure 24. Monitor - Evaluation Module Connections

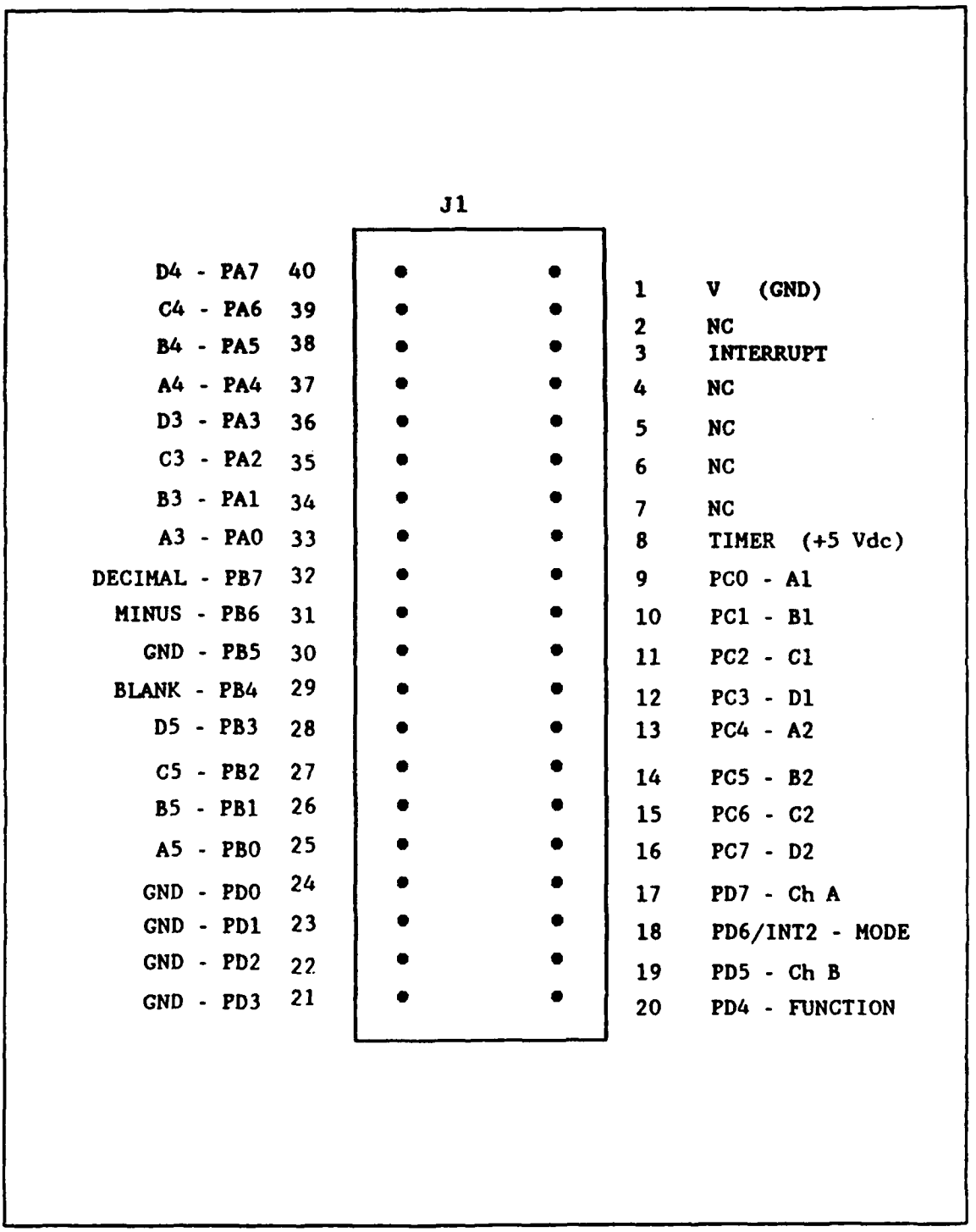


Figure 25. Evaluation Module - Signal Processor Connections

determining the angular distance through which the camera rotated between successive pulses from the shaft encoder. This number is a scale factor, which, when multiplied by the total number of pulses from the shaft encoder, yields a number equal to the angular displacement of the camera. The calibration procedure also involved determining the amount of hysteresis present in each of the gear trains.

1. **Scale Factor**

To determine the scale factor (SF) the simple geometric relationship

$$\theta = \tan^{-1}\left(\frac{\text{opposite}}{\text{adjacent}}\right) \tag{12}$$

was used. Using a small laser attached to the camera servo, and the geometry shown in Figure 26, the SF could be experimentally determined. As the camera servo was rotated through an angle, θ, the MPU was used to count the output pulses from the shaft encoder. The laser beam was projected on a vertical surface at a distance, a, away from the axis of rotation. The beam of the laser spread to a diameter of approximately 0.4 in. over a distance of 30 ft. A template with a 0.4 in. diameter aperture was used to mark the location of the "spots" on the distant wall. The distance between the spots, l, was measured by selecting one edge of one of the marks and measuring the distance to the corresponding edge of the distant mark. Then, having determined l and a and reading the count, C, from the display, the SF could then be determined from

$$\begin{aligned} \text{SF(degrees/Pulse)} &= \frac{\theta}{\text{C}} \\ &= \frac{\tan^{-1}\left(\frac{l(\text{in})}{a(\text{in})}\right)}{\text{C}} \end{aligned} \tag{13}$$

Using Equation (13) to simplify the expression,

$$\text{d(SF)} = \frac{\delta(\text{SF})}{\delta l}\,\text{d}l + \frac{\delta(\text{SF})}{\delta a}\,\text{d}a \tag{14}$$

yields

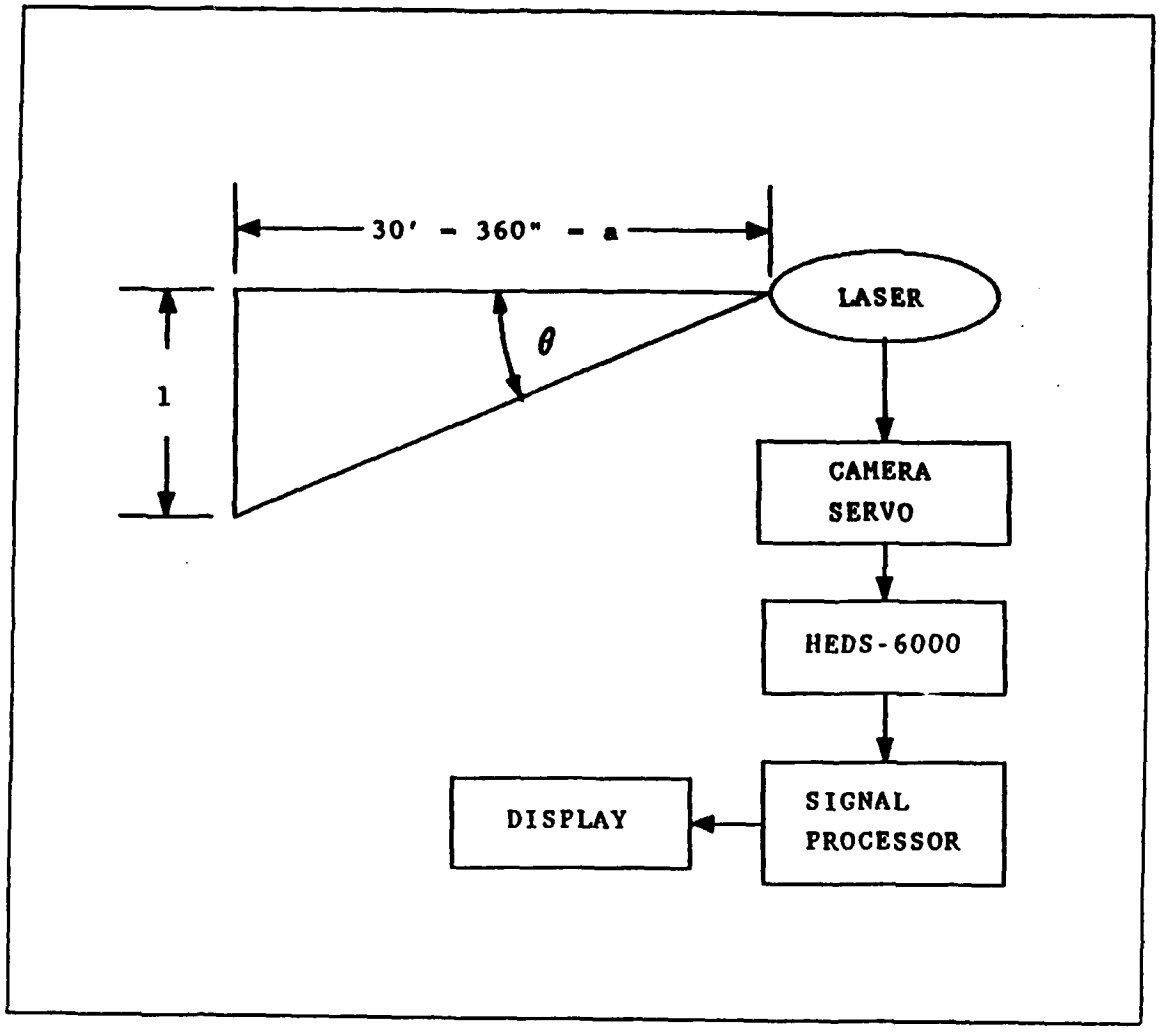


Figure 26. Geometry Used to Determine the Scale Factor

$$
\begin{aligned}
d(SF) &= \frac{1}{C}\frac{\delta\theta}{\delta l} + \frac{1}{C}\frac{\delta\theta}{\delta a}\,da \\
&= \frac{1}{C}\left(\frac{a}{a^2+l^2}\right)dl + \frac{1}{C}\left(-\frac{l}{a^2+l^2}\right)da \qquad (15) \\
d(SF) &= \frac{(a\,dl - l\,da)}{C(a^2+l^2)} \quad ,
\end{aligned}
$$

which indicates that C, a and l should all be as large as possible to minimize the error in SF due to a measurement error in a or l. The physical size of the laboratory limited the distance, a , to 30 ft. When $a = 30$ ft, l was limited to about 3.5 ft in the horizontal

plane and about 4.0 ft in the vertical plane. By modifying the geometry as shown in Figure 27, the count, which from Equation (13) is directly proportional to θ, could also be maximized. The configuration shown in Figure 27 was used to obtain the scale factor calibration data for the Pan axis. However, since the servo is incapable of rotating 360° about the Tilt axis, the test configuration shown in Figure 26 had to be used the collect the data for that axis.

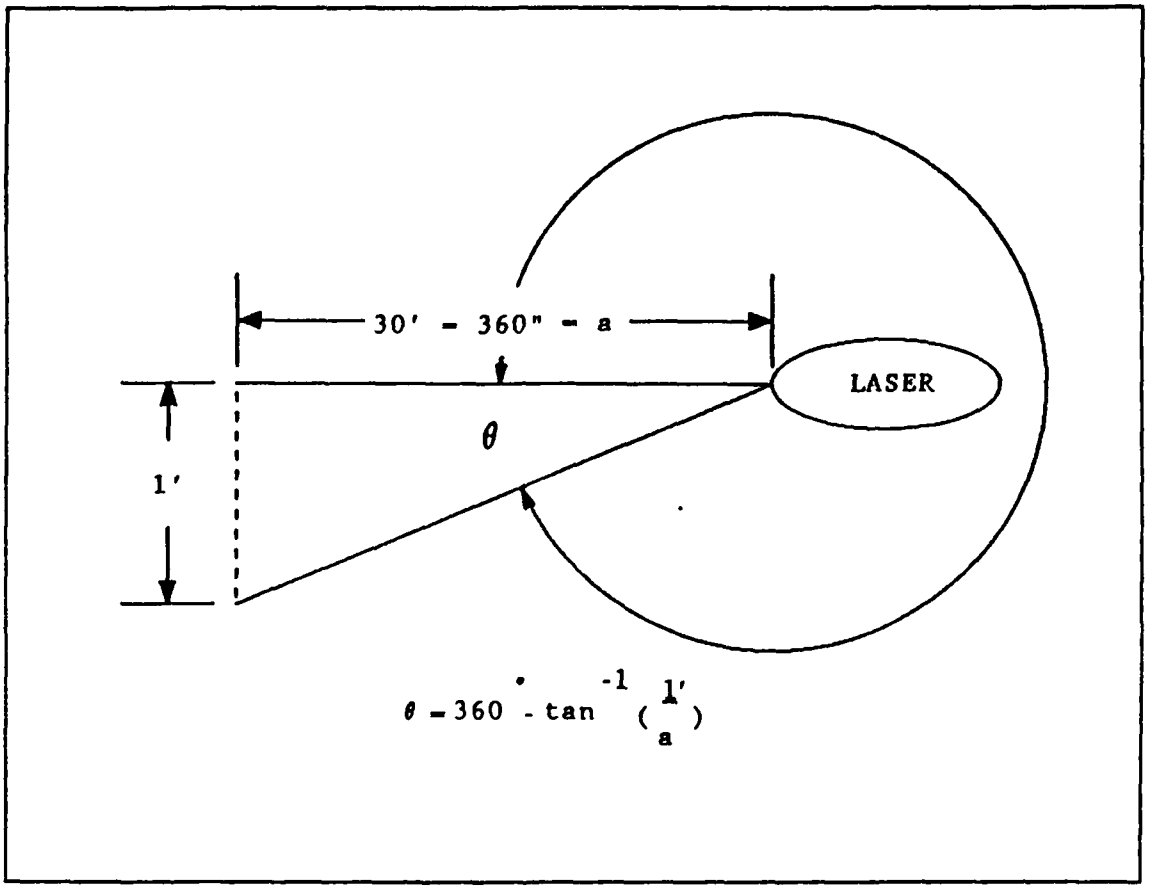


Figure 27. Alternate Geometry Used to Determine the Scale Factor

Adopting the notation, $\overline{X}$, to represent the mean value of a random variable, X; if X is discrete with N measured values,

$$\overline{X} \simeq \frac{1}{N} \sum_{i=1}^{N} X_i \tag{16}$$

where X_i is the ith measured value of X, and the approximation becomes better as N approaches infinity.

Using the expression in Equation (16) and the measured data for the scale factors, from 31 measurements on the Pan axis,

$$\overline{SF}_{Pan} = (7.0312 \times 10^{-3})°Pulse^{-1} \ , \tag{17}$$

and after 32 measurements on the Tilt axis,

$$\overline{SF}_{Tilt} = (7.0452 \times 10^{-3})°Pulse^{-1} \ , \tag{18}$$

where the subscripts indicate the axis. The actual implementation of these scale factors is described later in this chapter and in the documentation for each of the MPU programs.

If the error in the ith measurement is described by

$$e_i = X_i - \overline{X} \ , \tag{19}$$

then the root mean square (RMS) error in N measurements of X is given by;

$$\sigma_x = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (X_i - \overline{X})^2} \ . \tag{20}$$

Note that this is also the definition of the standard deviation of X.

The RMS errors in the Pan and Tilt scale factor measurements were determined from Equation (20) and the measured data to be;

$$\sigma_{SF_{Pan}} = (4.62 \times 10^{-6})°Pulse^{-1} \tag{21}$$

$$\sigma_{SF_{Tilt}} = (6.89 \times 10^{-6})°Pulse^{-1} \ . \tag{22}$$

As before, the subscripts are used to identify the axis and the source of the error. The fact that the errors are small compared to the mean values suggests that the means should closely approximate the actual values for the scale factors.

2. Hysteresis

Houghton defines backlash in wormgearing as "...the total play between the surfaces of the worm and worm wheel teeth measured normal to the faces." [Ref. 19: pp. 1.4, 1.5] Backlash only poses a problem in the measurement system when the servo's

direction of rotation changes. Figure 28 is a typical hysteresis curve. As long as the direction of rotation of the worm is increasing the $\theta_{WORM}/\theta_{WORMGEAR}$ relationship is linear. However, when the direction of rotation reverses there is a region, depicted by the left pointing arrows, where the position of the worm changes without a corresponding change in the position of the wormgear. Note that once all of the backlash has been taken up the $\theta_{WORM}/\theta_{WORMGEAR}$ relationship is again linear until the direction of rotation changes.

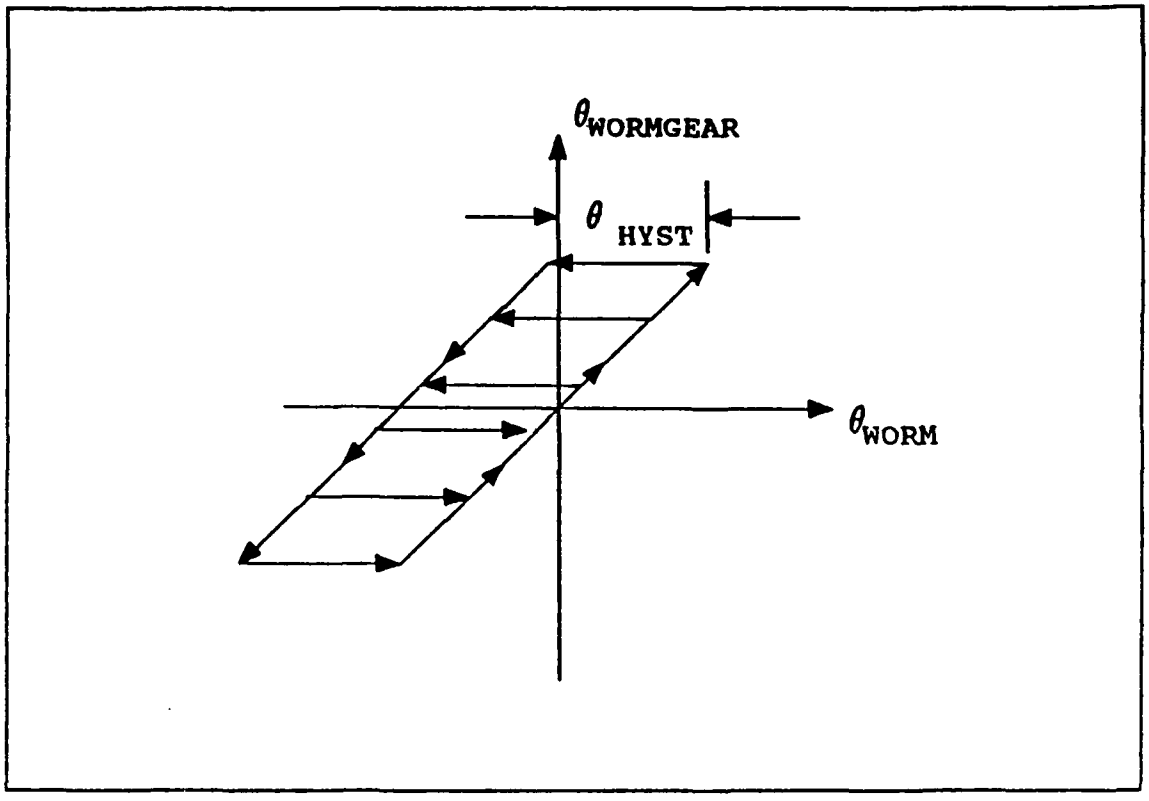


Figure 28. Typical Hysteresis Curve

The purpose of the hysteresis buffer in the MPU is to permit the signal processor to account for the backlash error introduced into the measurement by the worm-wormgear connection. The theory of operation for the buffer is relatively straight forward and is best described by the flow diagram in Figure 29. The buffer is a data byte in the MPU RAM. As long as the buffer is full, i.e., the contents are equal to the predetermined buffer length, a clockwise (CW) signal from the shaft encoder (increasing elevation and increasing azimuth are defined as CW rotation for the purposes of this system) causes the position counter to be incremented. Similarly, counter-clockwise

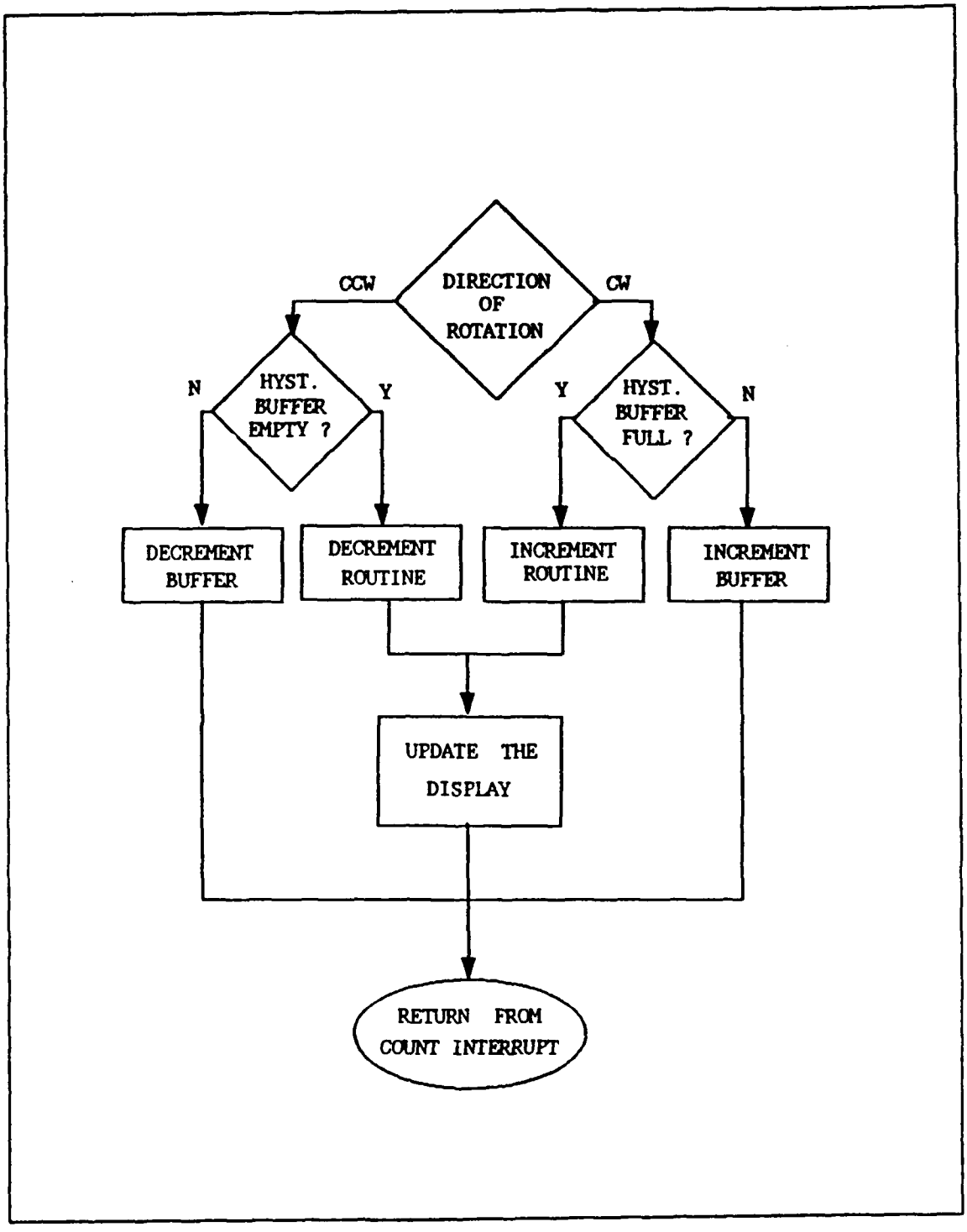


Figure 29. Operation of the Hysteresis Buffer

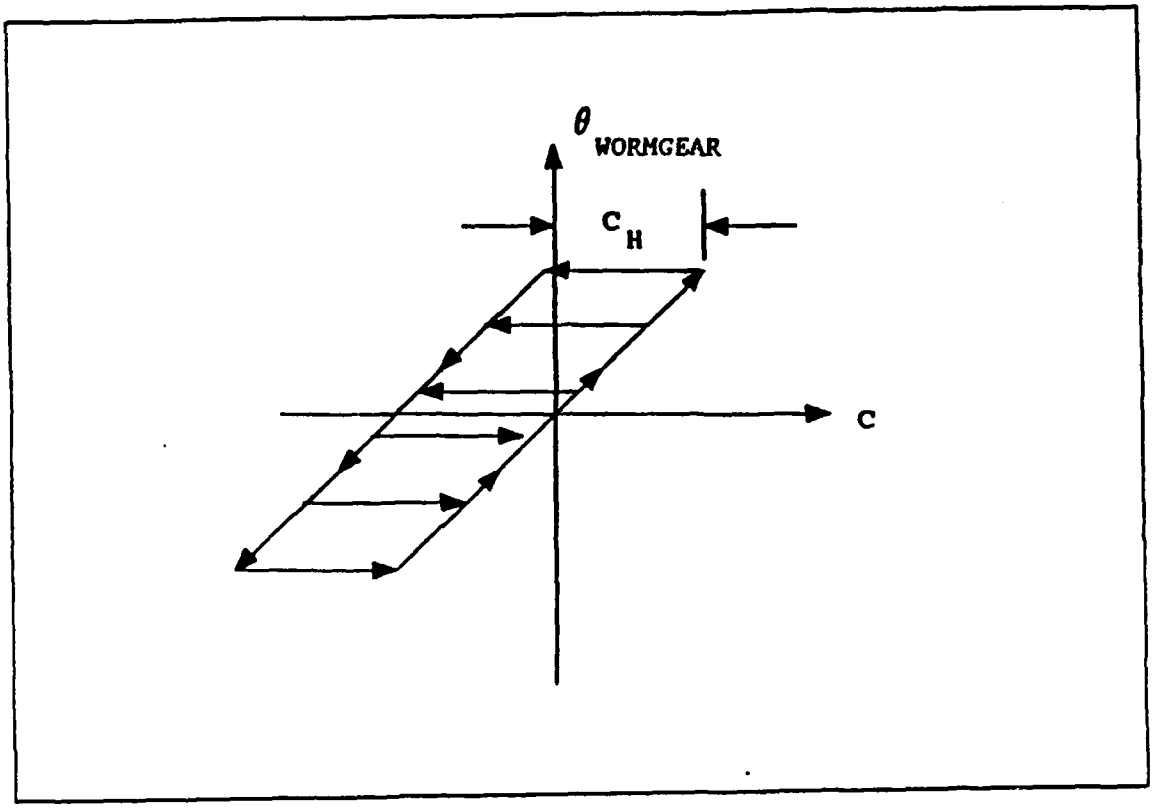


Figure 30. Hysteresis Curve

(CCW) signals cause the position counter to be decremented if the hysteresis buffer is empty, i.e., the contents are equal to zero. These two situations correspond to the two linear sections in Figure 28. From Equation (13), the horizontal separation of these two lines is related to the length of the hysteresis buffer, C_H by the expression

$$C_H = \frac{\Delta\theta_H}{\overline{SF}} \quad . \tag{23}$$

Similarly, using the $\overline{SF}$ to map θ_{WORM} into C, and the fact that the displacement of the wormgear equals the displacement of the axis of interest, the curve shown in Figure 30 can be obtained from Figure 28. From Figure 30 it is apparent that two different pulse counts can be obtained for any given position, θ, depending on whether that position is approached from a CW or a CCW direction. The difference in the two counts is a measure of the hysteresis present in the gear train and is also the required length for the hysteresis buffer. By using this difference as the length of the hysteresis buffer, counts received by the MPU which occur while the gears are operating on one of the horizontal

sections of the curve in Figure 30 are not considered "valid" and therefore do not cause the MPU to modify the position. Using Equation (23) and data collected in the laboratory the average hysteresis present in each of the gear trains was determined (from 45 measurements on the Pan axis and 30 measurements on the Tilt axis) to be

$$\overline{C}_{H(Pan)} = 7.39 \text{ Pulses} \tag{24}$$

$$\overline{C}_{H(Tilt)} = 6.06 \text{ Pulses} \tag{25}$$

and the RMS errors were calculated to be,

$$\sigma_{C_{H(Pan)}} = 1.0 \text{ Pulses} \tag{26}$$

$$\sigma_{C_{H(Tilt)}} = 0.91 \text{ Pulses} . \tag{27}$$

C. IMPLEMENTATION OF THE CALIBRATION DATA

1. Background

Figure 31 outlines the basic operation of each of the MPU's. Although each of the routines is described in detail by the comments included in the programs, the operation of the Count Routine is the heart of the program and should be explained prior to discussing the actual implementation of the experimental results.

When the system operator causes the camera servo to rotate, each optical shaft encoder translates the displacement of one of the axes into two series of digital pulses. The two pulse trains, referred to as Channels A and B, are TTL logic level signals. When the logic level of Channel A transitions from low to high (rising edge transition) or from high to low (falling edge transition), the edge detector (See Figure 15 on page 33) pulls pin 3 of the associated MPU low for approximately 2 μsec. When this occurs an external interrupt (EXT INT or $\overline{\text{INT}}$) request is generated and the MPU begins execution of the Count Routine.

As mentioned in Chapter III, since both rising and falling edge transitions are detected by the edge detector, the signal processor must be capable of detecting multiple oscillations of the shaft about a single logic level transition point. Accordingly, the first tasks performed by the Count Routine are to determine the direction of rotation and to simultaneously determine whether the interrupt is the result of a stationary shaft oscillation. To do this the Count Routine checks the state of pin 17 (Channel A) and pin 19 (Channel B). Operation of this portion of the routine is summarized in Table 5.

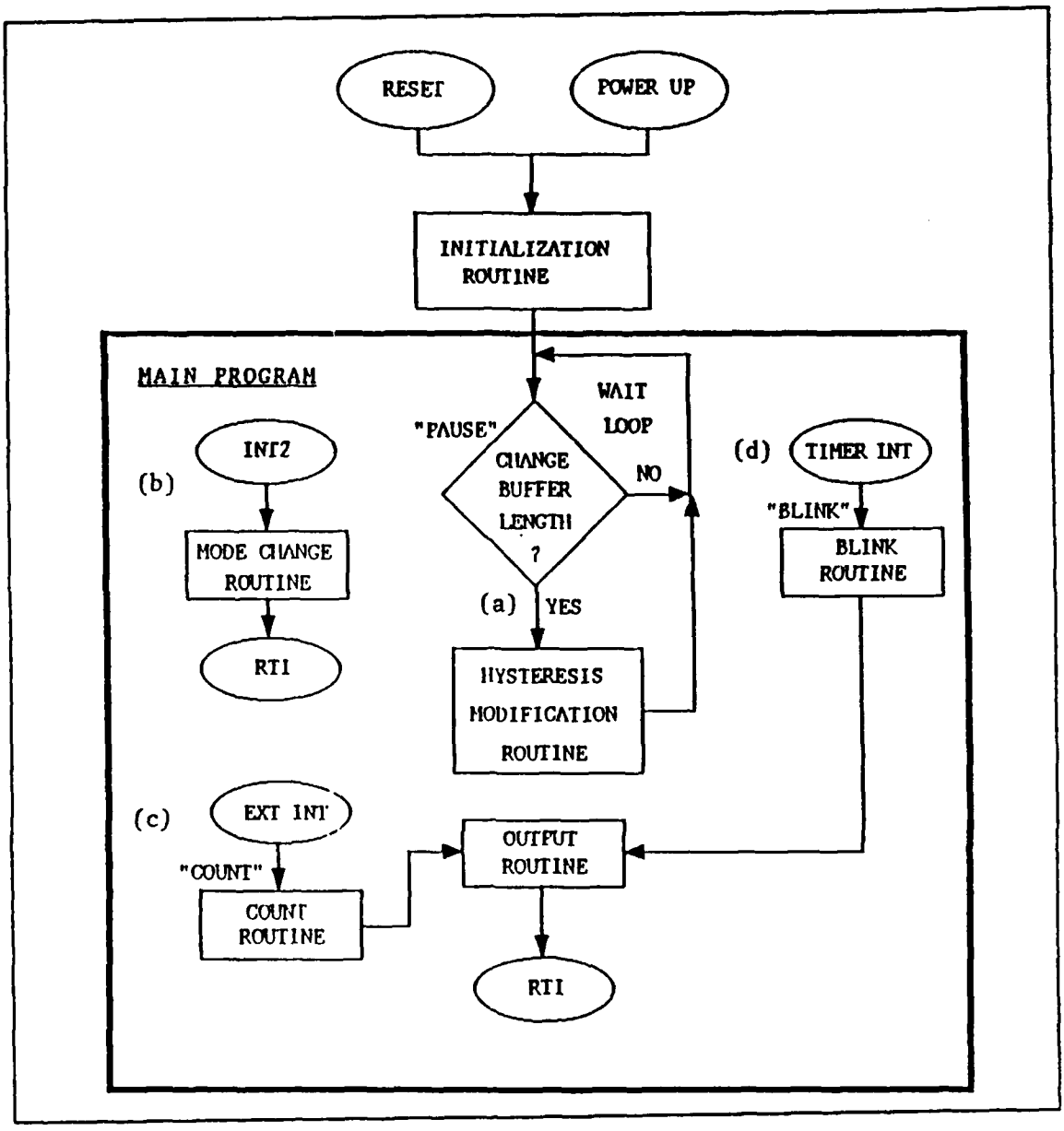


Figure 31. Program Flow Diagram: Program executes in the Wait Loop until; (a) the operator requests to modify the hysteresis buffer, (b) the operator requests to change the display mode, (c) an external interrupt is generated by the Channel A edge detector or, (d) a timer interrupt causes the display mode to "blink". All interrupt routines are terminated with a return from interrupt (RTI) command.

Note that CW rotation is indicated when Channel B leads Channel A in phase and CCW rotation is indicated if Channel A is leading Channel B. The possibility of erroneously counting multiple oscillations about a single point is eliminated by "counting" only the leading edge transitions when the shaft is rotating CW and only the trailing edge transitions when the rotation is CCW. All other transitions cause the program to execute a "return from interrupt" (RTI) instruction.

2. Implementing the Hysteresis Buffer

The transitions that are to be counted cause the program to compare the contents of the hysteresis buffer, HYSTCT, and the direction of rotation to the experimentally determined buffer length, HYST$\simeq\overline{C}_H$. If the rotation is CW and the buffer is full (i.e., HYSTCT = HYST), or if the rotation is CCW and the buffer is empty (i.e., HYSTCT = 0), then the "slack" due to the gear backlash should have been taken up , the transitions are considered "valid" and the MPU modifies the position appropriately. "Invalid" counts cause the contents of the hysteresis buffer to be incremented or decremented depending on whether the present direction of rotation is CW or CCW (Figure 29 refers).

Table 5. COUNT ROUTINE LOGIC

Channel A	Channel B	Direction of Rotation	Count the Pulse?	Action
Low	Low	CW	No	Increment the hysteresis buffer.
Low	High	CCW	Yes	Decrement the position.
High	Low	CCW	No	Decrement the hysteresis buffer.
High	High	CW	Yes	Increment the position.

From the calibration data for the Pan axis, $\overline{C}_{H(Pan)}\simeq 7.4$ and $\sigma_{C_{H(Pan)}}\simeq 1.0$. Because the length of the hysteresis buffer must be an integer value, $\overline{C}_{H(Pan)}$, needed to be rounded off. Rounding to the nearest whole number initially seemed the most logical approach. Upon further consideration, however, it was decided to round 7.4 up to 8. Considering the relatively small data base (45 measurements) upon which the average was based, the fact that the standard deviation was 1.0 and that the gear backlash will only increase

with time, this seemed like the most reasonable approach. The buffer length for the Tilt axis was set equal to 6 ($\overline{C}_{H(Tilt)} = 6.06$) .

3. Implementing the Scale Factor

There are three counters in each MPU that keep track of the position information for the axis of interest. The first, BINCT, is simply a binary counter that is incremented by one for each valid CW count and decremented by one for each valid CCW count. The other two counters consist of two sets of pointers, two sets of data registers and a shared data table. Each byte in a pointer points to an address in the table that contains two BCD digits which make up a portion of the position information.

In order to increment (decrement) the pulse count, BCDCT, by one, the count pointer, CTPTR, is incremented (decremented) by one causing it to point to a new table address. The contents of the table at the new addresses are then moved into BCDCT. Modification of the position counter, DEGRES, is performed in much the same manner. DEGRES contains a BCD number that, when multiplied by 0.001, represents the angular position (in degrees) of the shaft of interest. Thus, each time the camera is displaced in a CW direction through one degree, the contents of DEGRES should be incremented by 1000. To do this, the position pointer, PTR, must be incremented (decremented) by seven or eight each time CTPTR is incremented (decremented) by one. Incrementing PTR by seven corresponds to an angular displacement of 0.007° and since $\overline{SF}_{Pan}$ and $\overline{SF}_{Tilt}$ are each slightly larger than 0.007°, periodically PTR must be incremented by eight to reduce the cumulative round off error. Specifically, if PTR is incremented by seven each time a valid CW pulse is detected (except when BINCT is an even multiple of 32) and is incremented by eight when BINCT is a modulo 32 number, the effective scale factor is given by;

$$\begin{aligned} SF_{eff_1} &= \frac{1}{32}(31(0.007) + 0.008) \\ &= 0.00703125°\text{Pulse}^{-1} \ , \end{aligned} \tag{28}$$

which is slightly larger than the desired 0.0070312°Pulse^{-1}, for the Pan axis and slightly smaller than the desired 0.0070452°Pulse^{-1} for the Tilt axis. To further reduce the cumulative round off on the Pan axis, every 16,384 ($2^{14} = 32 \times 512$) counts PTR is incremented by seven instead of eight. This results in an effective scale factor of,

$$SF_{eff_2} = \frac{1}{512}\left\{511\left[\frac{1}{32}(31(7)+8)\times 10^{-3}\right] + 0.007\right\}$$
$$= \left(7.03119 \times 10^{-3}\right)^{\circ}\text{Pulse}^{-1} \ , \tag{29}$$

which is within two parts in one million.

The maximum error introduced into the measurement should occur when the camera is rotated through the largest possible angle. To predict this error on the Pan axis, when the camera has been displaced by 360°,

$$BCDCT = \frac{360^{\circ}}{\overline{SF}_{Pan}}$$
$$= 51{,}200 \text{ Pulses} \tag{30}$$

so the position error using SF_{eff_1} and SF_{eff_2} should be;

$$e_{Pan} = 360^{\circ} - [3(SF_{eff_2}) + (51{,}200 - 3(16{,}384))(S_{eff_1})]$$
$$= 0.00295^{\circ} \ . \tag{31}$$

Thus, the error due to the scale factor on the Pan axis should be well within the desired resolution of $\pm 0.006^{\circ}$.

Using SF_{eff_1} alone as the scale factor for the Tilt axis, the maximum theoretical error over $\pm 12^{\circ}$ due to the scale factor round off is determined in the same manner.

$$BCDCT = \frac{12^{\circ}}{\overline{SF}_{Tilt}}$$
$$= 17{,}033 \text{ Pulses} \tag{32}$$

So that,

$$e_{Tilt} = 12^{\circ} - 17{,}033(SF_{eff_1})$$
$$= 0.0234^{\circ} \ , \tag{33}$$

which again is significantly less than the required resolution of $\pm 0.23^{\circ}$ for the Tilt axis.

The final step in the Count Routine is a branch to the Display Routine. Depending on whether the MPU is in the Count or Position Mode, the Display Routine copies the contents of BCDCT or DEGRES to the output ports and then executes an RTI instruction.

D. FINAL TESTING

1. General

Final laboratory testing and evaluation of the measurement system was performed after the calibration results had been implemented in each of the MPU Count Routines. The purpose of the testing was to verify that the calibration results had been properly coded into the MPU's and to determine the resolution capabilities of the measurement system experimentally.

This verification process included determining the combined error due to the hysteresis and scale factor errors. The use of some simple multiple random variable theory was therefore required. From [Ref. 20: pp. 121,122] the variance of a weighted sum of M random variables is the weighted sum of their covariances, $C_{X_iX_j}$, and is given by,

$$\sigma^2 = \sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i\alpha_j C_{X_iX_j} \ , \tag{34}$$

where α_i is the weight associated with X_i. Additionally, the covariance can be expressed as

$$C_{XY} = \rho\sigma_X\sigma_Y \ , \tag{35}$$

where ρ is the normalized second-order moment and is known as the correlation coefficient of X and Y. The correlation coefficient is bounded by

$$-1 \le \rho \le 1 \ . \tag{36}$$

In the case where there are two equally weighted random variables, $M = 2$ and $\alpha_i = \alpha_j = 1.0$. Substituting into Equation (34) and expanding

$$\sigma^2 = C_{XX} + C_{XY} + C_{YX} + C_{YY} \ . \tag{37}$$

Using Equation (35),

$$\sigma^2 = \sigma_X^2 + 2\rho\sigma_X\sigma_Y + \sigma_Y^2 \ . \tag{38}$$

Combining (36) and (38) yields

$$\sigma_X^2 - 2\sigma_X\sigma_Y + \sigma_Y^2 \le \sigma^2 \le \sigma_X^2 + 2\sigma_X\sigma_Y + \sigma_Y^2 \ . \tag{39}$$

The bounds of the combined scale factor and hysteresis errors can therefore be determined from

$$\sqrt{\sigma_{e_{SF}}^2 - 2\sigma_{e_{SF}}\sigma_{e_H} + \sigma_{e_H}^2} \leq \sigma_e \leq \sqrt{\sigma_{e_{SF}}^2 + 2\sigma_{e_{SF}}\sigma_{e_H} + \sigma_{e_H}^2} \quad . \tag{40}$$

where:

- σ_e = standard deviation of the combined error,
- $\sigma_{e_{SF}}$ = standard deviation of the scale factor error, and
- σ_{e_H} = standard deviation of the hysteresis error.

Note that the bounds are determined by the two cases where the hysteresis error, e_H, and the scale factor error, e_{SF}, are "completely correlated". The upper bound corresponds to the case where an increase in e_{SF} implies an increase in e_H, and the lower bound corresponds to the case where an increase in e_{SF} directly implies a decrease in e_H.

A third case is also of particular interest. If e_H and e_{SF} are completely uncorrelated, i.e., $\rho = 0$, then from Equation (38),

$$\sigma_e = \sqrt{\sigma_{e_{SF}}^2 + \sigma_{e_H}^2} \quad , \tag{41}$$

which can also be written as

$$e = \pm \sqrt{e_{SF}^2 + e_H^2} \quad , \tag{42}$$

where $e = \pm \sigma_e$ is the 1.0 σ error due to the scale factor and hysteresis errors, e_{SF} and e_H respectively. Based on the physical nature of the two errors it is reasonable to assume that e_{SF} and e_H are statistically uncorrelated; however, no experimental data was collected to support this hypothesis. Due to this lack of a priori information, the maximum RMS error, $e_{max} = \pm \sigma_{e_{max}}$, given by

$$e_{max} = \pm \sqrt{e_{SF}^2 + 2e_{SF}e_H + e_H^2} \quad , \tag{43}$$

will be used to describe the resolution capabilities of the measurement system.

2. Pan Axis

a. Hysteresis

To test the operation of the Pan axis hysteresis buffer, the Initialization Routine was programmed to set the buffer length to 8. Then, as described in the first section of this chapter, the servo was used to position the beam of a small laser on a fixed target. By approaching the target alternately from a CW and a CCW direction and

comparing the difference in the output counts from the shaft encoders, the hysteresis error was determined. The average error after 31 measurements was

$$\bar{e}_{H(Pan)} = -0.1880 \text{ Pulses} , \tag{44}$$

and the RMS error was

$$\sigma_{e_{H(Pan)}} = 1.7699 \text{ Pulses} . \tag{45}$$

Since the mean error is "near zero" compared to the standard deviation, the 1.0 σ error e_H, due to the hysteresis can be determined from

$$e_H = \pm (\sigma_{e_H})\overline{SF} \tag{46}$$

so that

$$e_{H(Pan)} = \pm 0.0124^\circ . \tag{47}$$

b. Scale Factor

Verification of the scale factor was performed in the same manner as the scale factor calibration, except that the MPU was calibrated in the Count Mode and tested in the Position Mode. Since these tests sought to find the maximum error due to the scale factor, and the error is directly proportional to the angle that is being measured, these tests were conducted by displacing the camera servo through the maximum angles permitted by the camera and the laboratory. Specifically, on the Pan axis the servo was rotated through approximately 360°. The mean error due to the scale factor on the Pan axis was determined from 15 samples to be

$$\bar{e}_{SF(Pan)} = -0.0018^\circ \tag{48}$$

and the RMS error was

$$\sigma_{e_{SF(Pan)}} = 0.00890^\circ . \tag{49}$$

As with the hysteresis error, if we neglect the small bias due to the $\bar{e}_{SF(Pan)}$, we can describe the 1.0 σ RMS error due to the scale factor as

$$\begin{aligned} e_{SF(Pan)} &= \pm \sigma_{SF(Pan)} \\ &= \pm 0.00890^\circ \end{aligned} \tag{50}$$

c. Combined Error

From Equation (42) the combined error on the Pan axis if e_H and e_{SF} are uncorrelated can be estimated as

$$\begin{aligned} e_{Pan} &= \pm\sqrt{(0.0124)^2 + (0.0089)^2} \\ &= \pm 0.01526^\circ \ . \end{aligned} \tag{51}$$

And from Equation (43) the maximum combined error on the Pan axis is

$$e_{Pan_{max}} = \pm\sqrt{(0.0124)^2 + 2(0.0124)(0.0089) + (0.0089)^2} \tag{52}$$

$$e_{Pan_{max}} = \pm 0.0213^\circ \ . \tag{53}$$

The combined error is approximately three times larger than the design specification limit and is due primarily to the hysteresis error.

3. Tilt Axis

a. Hysteresis

The procedure used to verify the operation of the calibrated Tilt MPU was identical to that described in the previous section. Using a hysteresis buffer length of 6 resulted in

$$\bar{e}_{H(Tilt)} = -0.0965 \text{ Pulses} \tag{54}$$

and,

$$\sigma_{e_{H(Tilt)}} = 1.3156 \text{ Pulses} \tag{55}$$

after 13 samples. As before we can define the 1.0 σ error from $e = \sigma\overline{SF}$ to be

$$e_{H_{Tilt}} = \pm 0.00529^\circ \ . \tag{56}$$

b. Scale Factor

Testing the scale factor on the Tilt axis was limited by the physical construction of the servo and the size of the laboratory. The angle over which testing could be performed was limited to $\pm 6^\circ$ from the horizontal plane. Consequently, the RMS error for the scale factor on the Tilt axis was determined in exactly the same manner as

the RMS error for the Pan axis scale factor, but since the error due to rounding of the scale factor is directly proportional to the angle being measured the results were multiplied by 2.0 to account for the limited range of the test. The modified results should therefore be representative of the maximum error one should expect if the measurement system is used to measure elevation angles over a range of $\pm 12°$.

The error due to the scale factor is described from 16 samples by

$$\bar{e}_{SF(Tilt)} = 0.0070° \tag{57}$$

and,

$$\sigma_{e_{SF(Tilt)}} = 0.0665° \ . \tag{58}$$

Including the factor of two in the calculation we have

$$\begin{aligned} e_{SF(Tilt)} &= \pm 2(0.0665°) \\ &= \pm 0.1330° \ . \end{aligned} \tag{59}$$

c. Combined Error

The combined RMS error is determined in the same manner as before. If e_H and e_{SF} are uncorrelated,

$$\begin{aligned} e_{Tilt} &= \pm \sqrt{(0.1330)^2 + (0.00529)^2} \\ &= 0.1331° \ , \end{aligned} \tag{60}$$

and the maximum combined RMS error is

$$e_{Tilt_{max}} = \pm \sqrt{(0.1330)^2 + 2(0.1330)(0.00529) + (0.00529)^2} \tag{61}$$

$$e_{Tilt_{max}} = \pm 0.1383° \ . \tag{62}$$

The Tilt axis measurement system appears to perform well within the required resolution specifications.

E. FINAL IMPLEMENTATION

Once the program debugging, calibration and testing were completed, final implementation of the system remained. Taking the design from the prototype/development model to a fully functional system was a straightforward but time-consuming evolution.

The plans for the printed circuit boards (PCB's) were made directly from the schematics shown in Appendix A; the boards were then etched and assembled from the plans which are shown in Appendix B. All of this work was performed by sailors attached to the Academic Division of the NPS.

As previously discussed, the M68705EVM Evaluation Module provided a powerful and flexible means of debugging and evaluating the performance of the microprocessor based signal conditioner. Additionally, once program testing was completed the EVM's EPROM microprocessor programmer provided the means to program the EPROM MCU's. A detailed, but simple to follow, programming procedure for programming the MC68705U3 is given in [Ref. 18: pp. (3-26)-(3-27)].

V. CONCLUSIONS AND RECOMMENDATIONS

A. SYSTEM PERFORMANCE

The prototype system was calibrated and successfully tested in a laboratory environment. Experimental results indicate that the system is capable of measuring the video camera's elevation over a range of $\pm 12°$ with a resolution of $\pm 0.138°$ and its azimuth over 360° with a resolution of $\pm 0.021°$. The system was designed to be low cost, reliable, and easy to operate. Only time will tell whether these objectives were truly achieved.

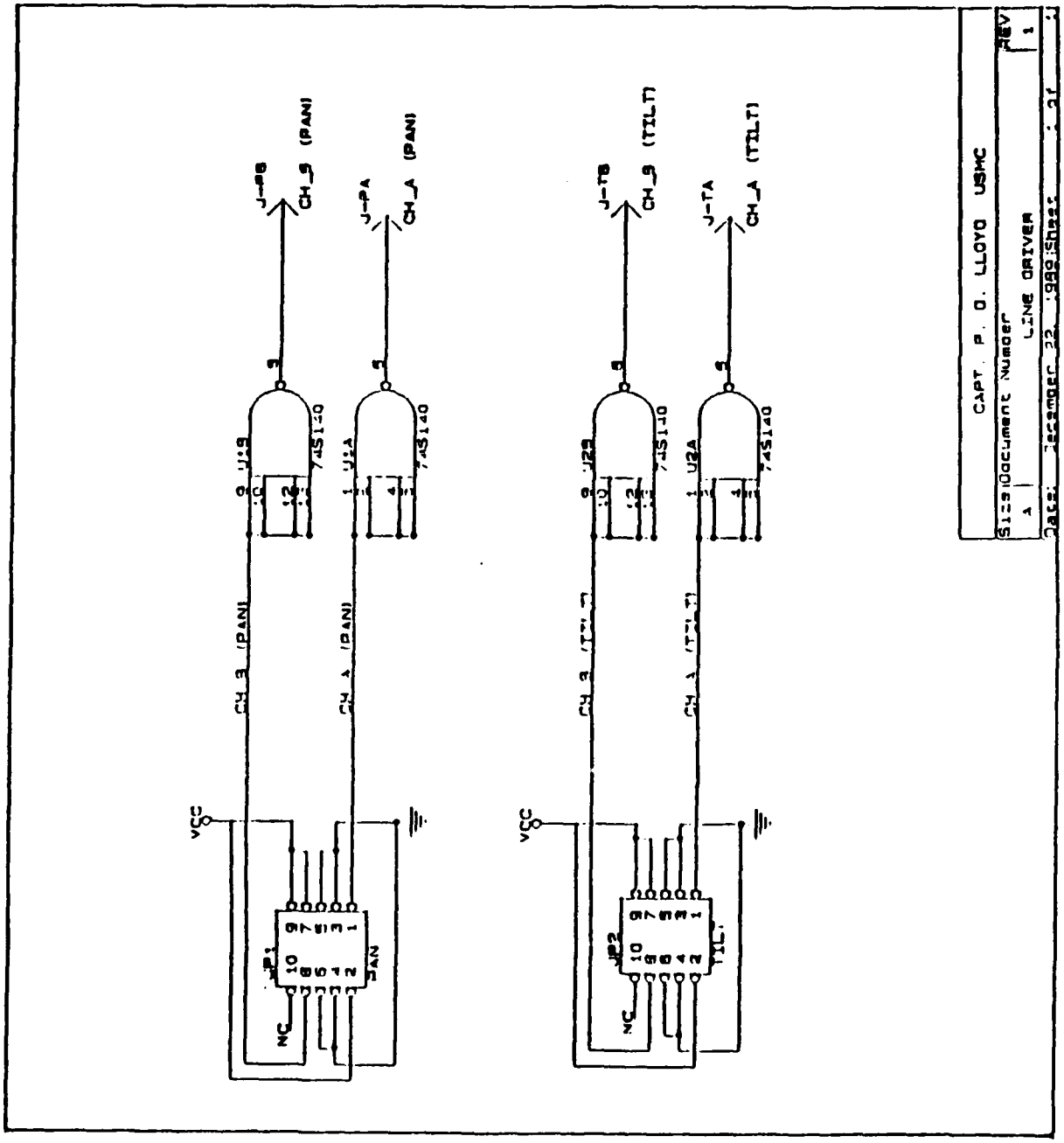
The portion of the system that will be located outdoors has been weatherproofed and is ready to be placed in service. Printed circuit board plans for the remainder of the system have been developed, but final implementation of the system is still ongoing. Once the system is fully operation additional testing should be performed in order to verify the completed system's performance. Although the laboratory results indicate that the system is capable of meeting all of the design criteria except for the required resolution on the Pan axis, the system must be further tested in a non-laboratory environment. "...The proof of a good design rests in the ability of the system to function in the outside world." [Ref 21]

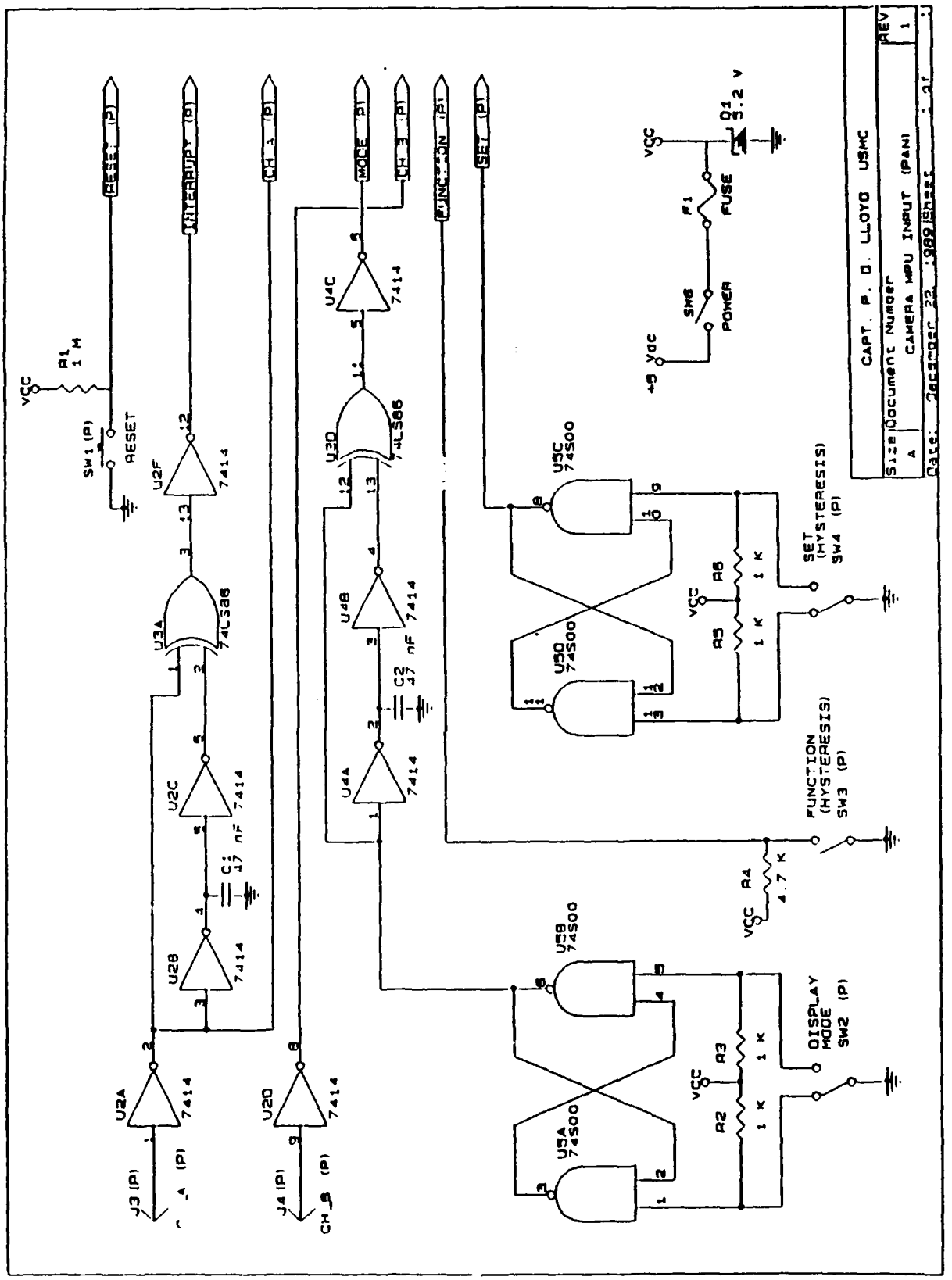
B. RECOMMENDATIONS FOR FURTHER WORK

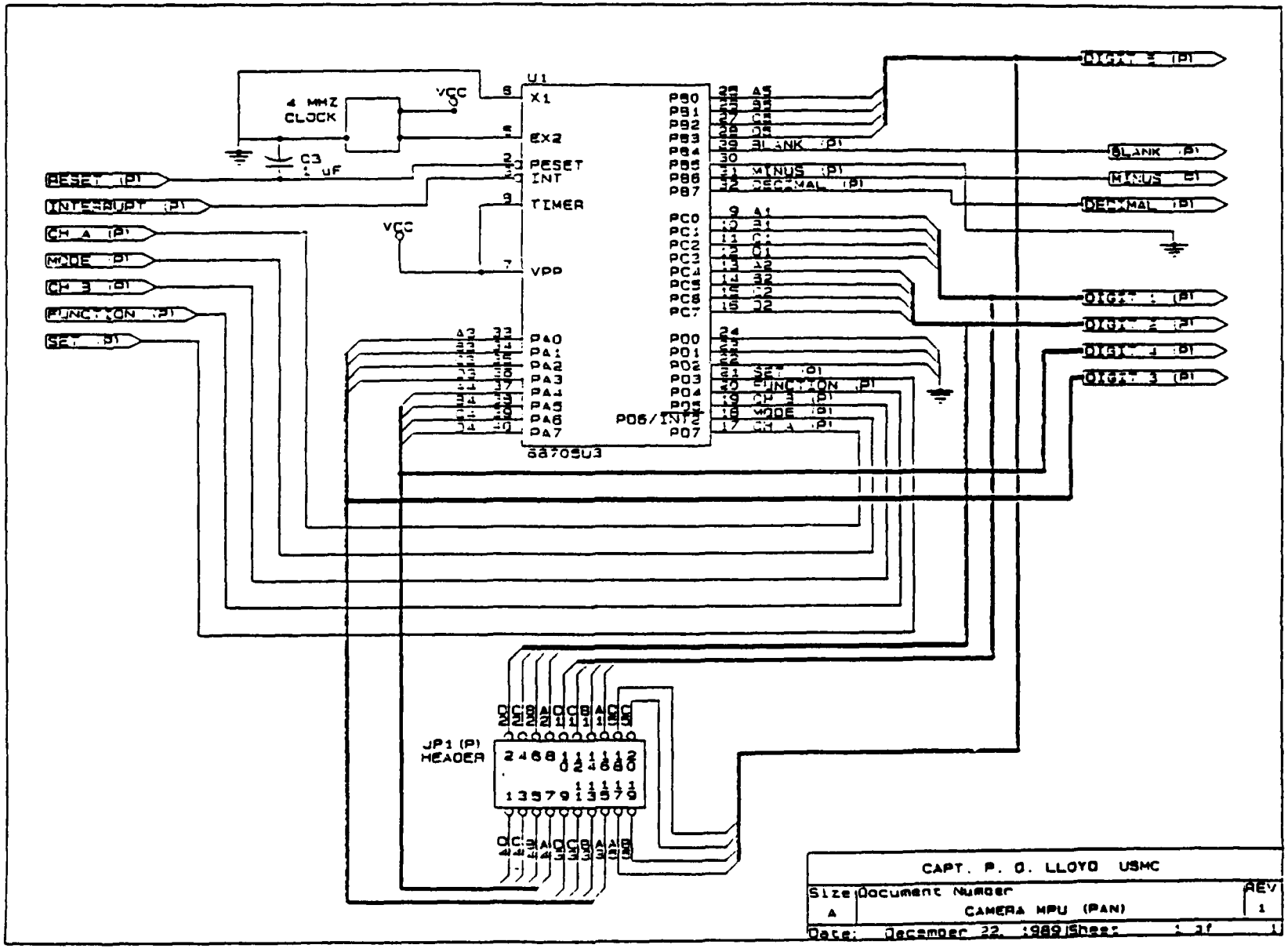
There are several areas for follow-on work with this project. Some possibilities are:

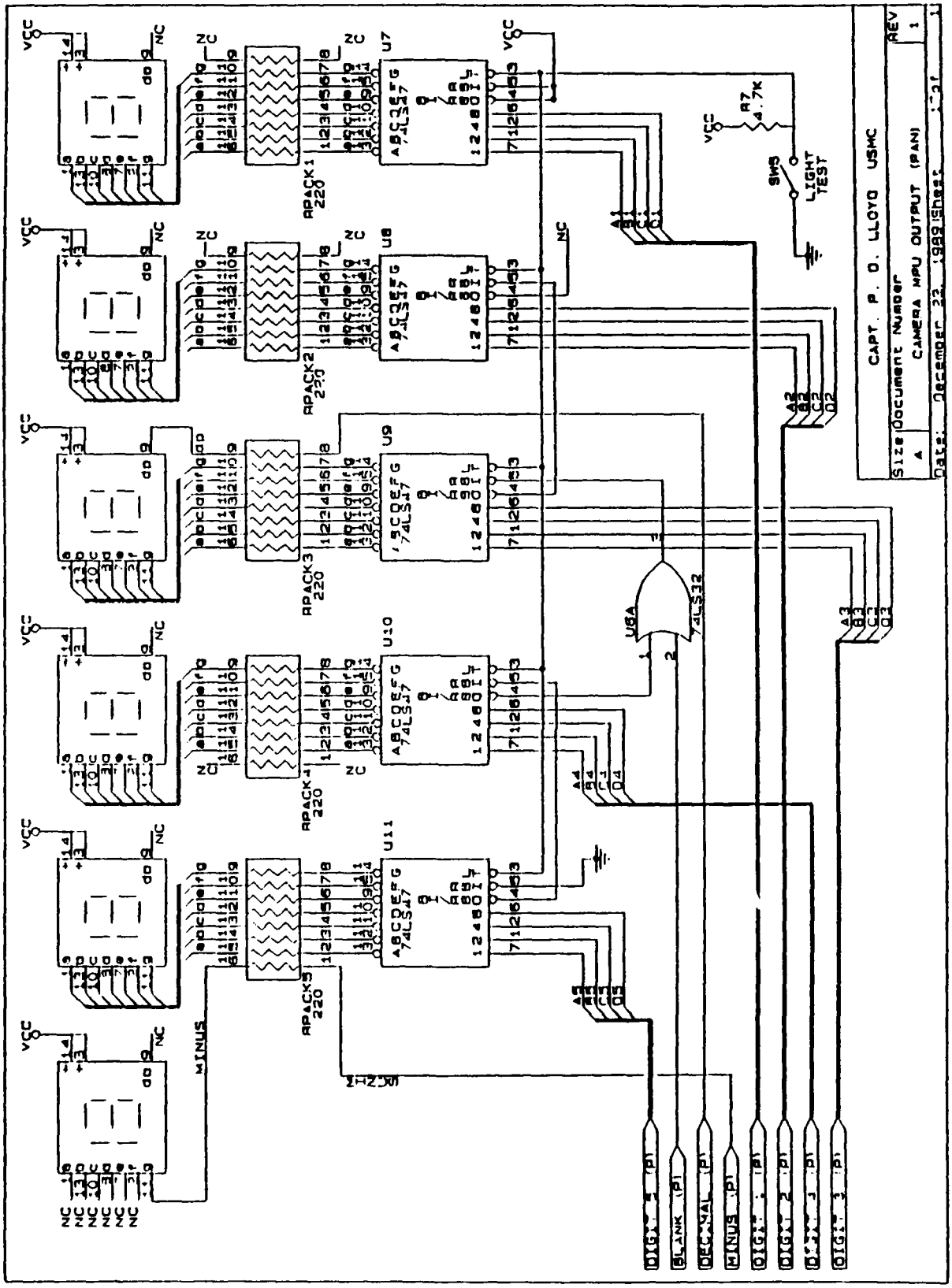
- Incorporate the measurement system's output into the video image being created by the video camera. This would permit a permanent record of the position information to be stored with the video image and would facilitate identification of the image at a later date.
- Design and build an automatic feedback control system for the camera.
- Implementation of a second video camera at the NPS, together with the NPS modified IRSTD, would permit triangulation of a target and would consequently provide range information which is not currently available. This information could provide valuable additional information to those who are developing the signal processing algorithms.

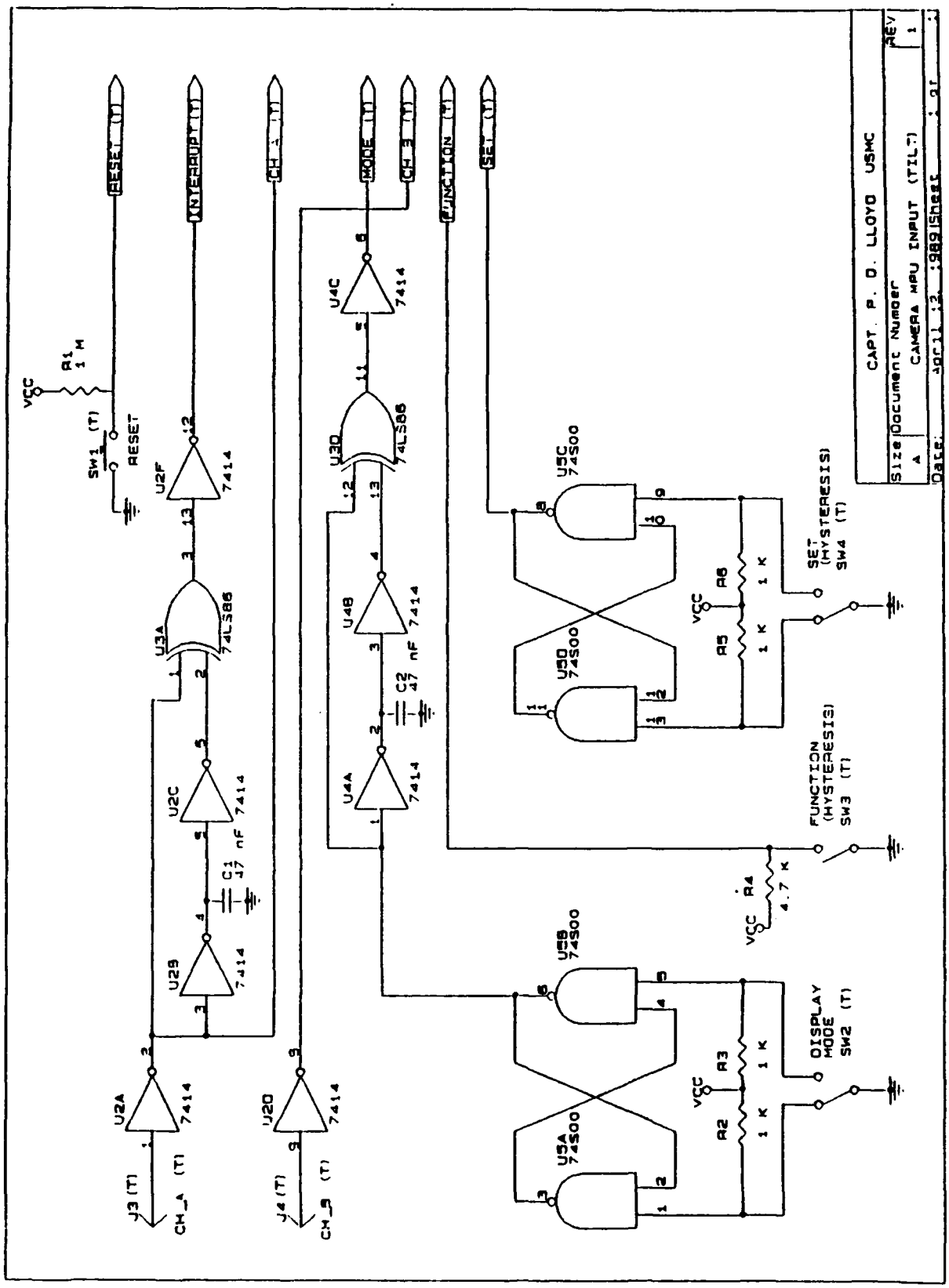
APPENDIX A. SCHEMATIC DIAGRAMS

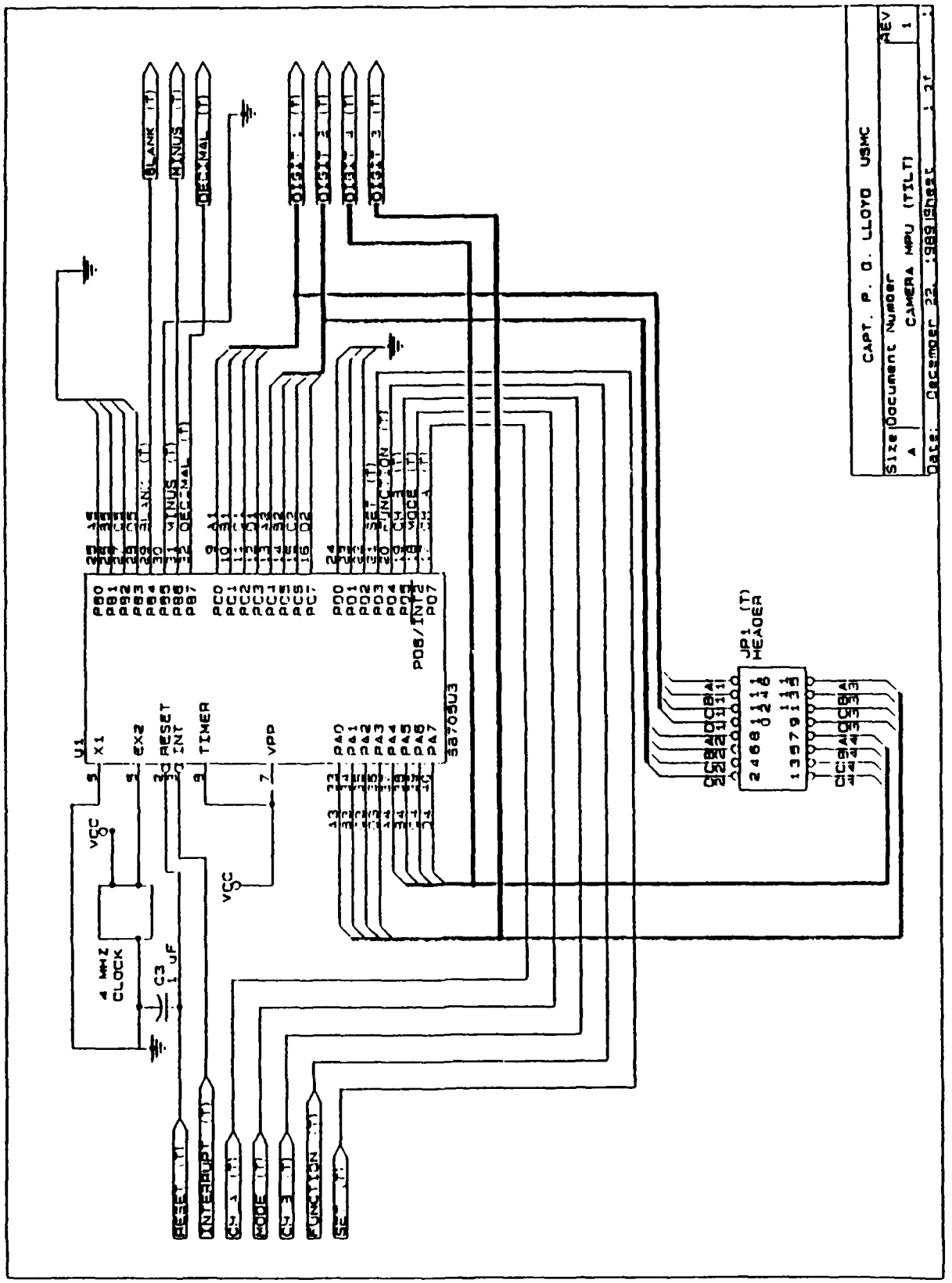


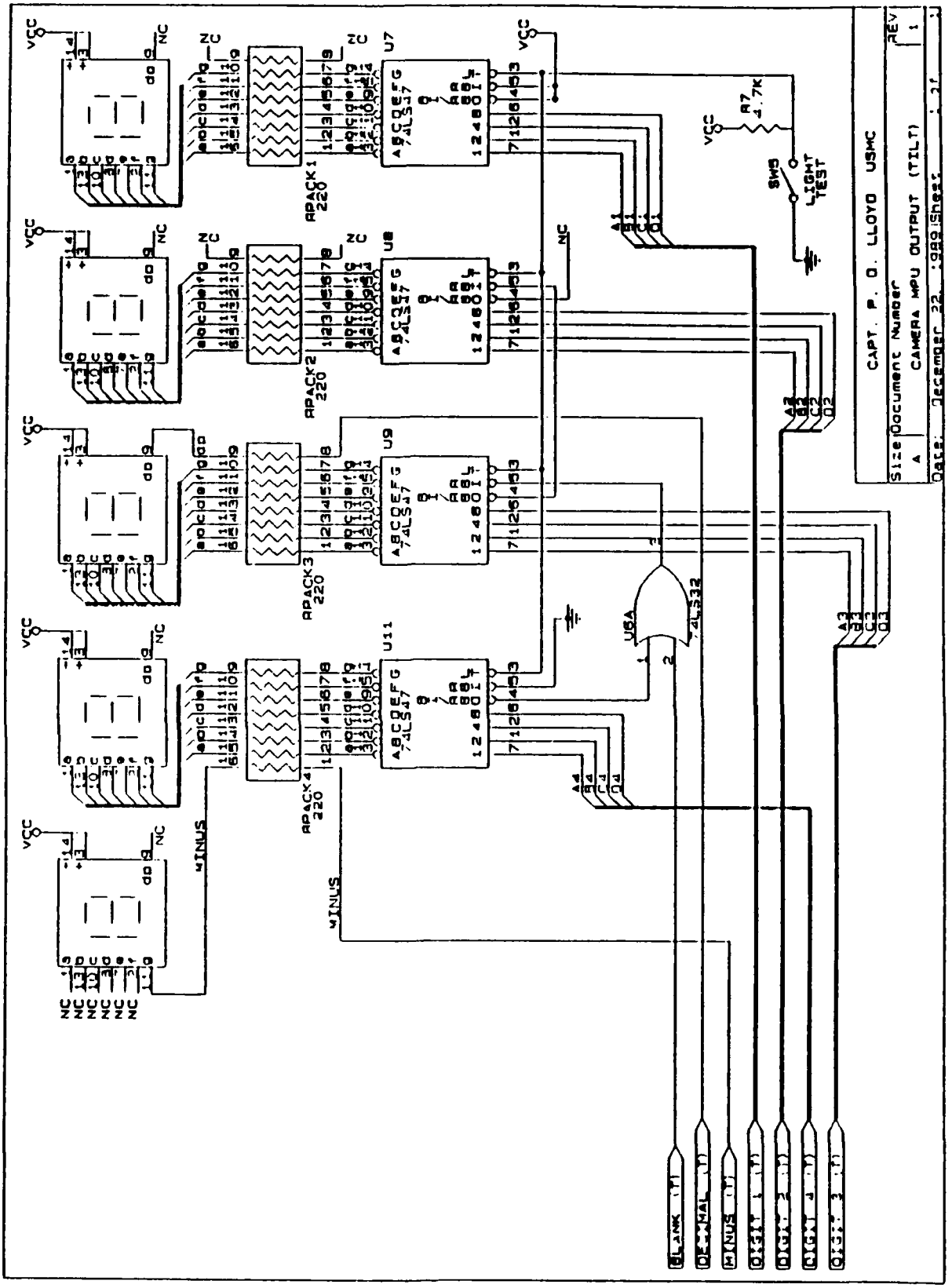

CAPT. P. D. LLOYD USMC
Size Document Number REV
A CAMERA MPU INPUT (PAN) 1
Date: December 22, 1989 Sheet of
RESET (P)
INTERRUPT (P)
CH 1 (P)
MODE (P)
CH 2 (P)
FUNCTION (P)
SET (P)
VCC
R1 1 M
SW1 (P)
RESET
U2F 7414
U3A 74LS86
U2C 7414
C1 47 nF
U2B 7414
U2A 7414
J3 (P)
A (P)
U2D 7414
J4 (P)
CH_B (P)
U4C 7414
U3D 74LS86
U4B 7414
C2 47 nF
U4A 7414
U5C 74S00
U5D 74S00
SET (HYSTERESIS) SW4 (P)
R6 1 K
R5 1 K
FUNCTION (HYSTERESIS) SW3 (P)
R4 4.7 K
U5B 74S00
U5A 74S00
DISPLAY MODE SW2 (P)
R3 1 K
R2 1 K
+5 VDC
SW8
POWER
F1
FUSE
Q1 5.2 V

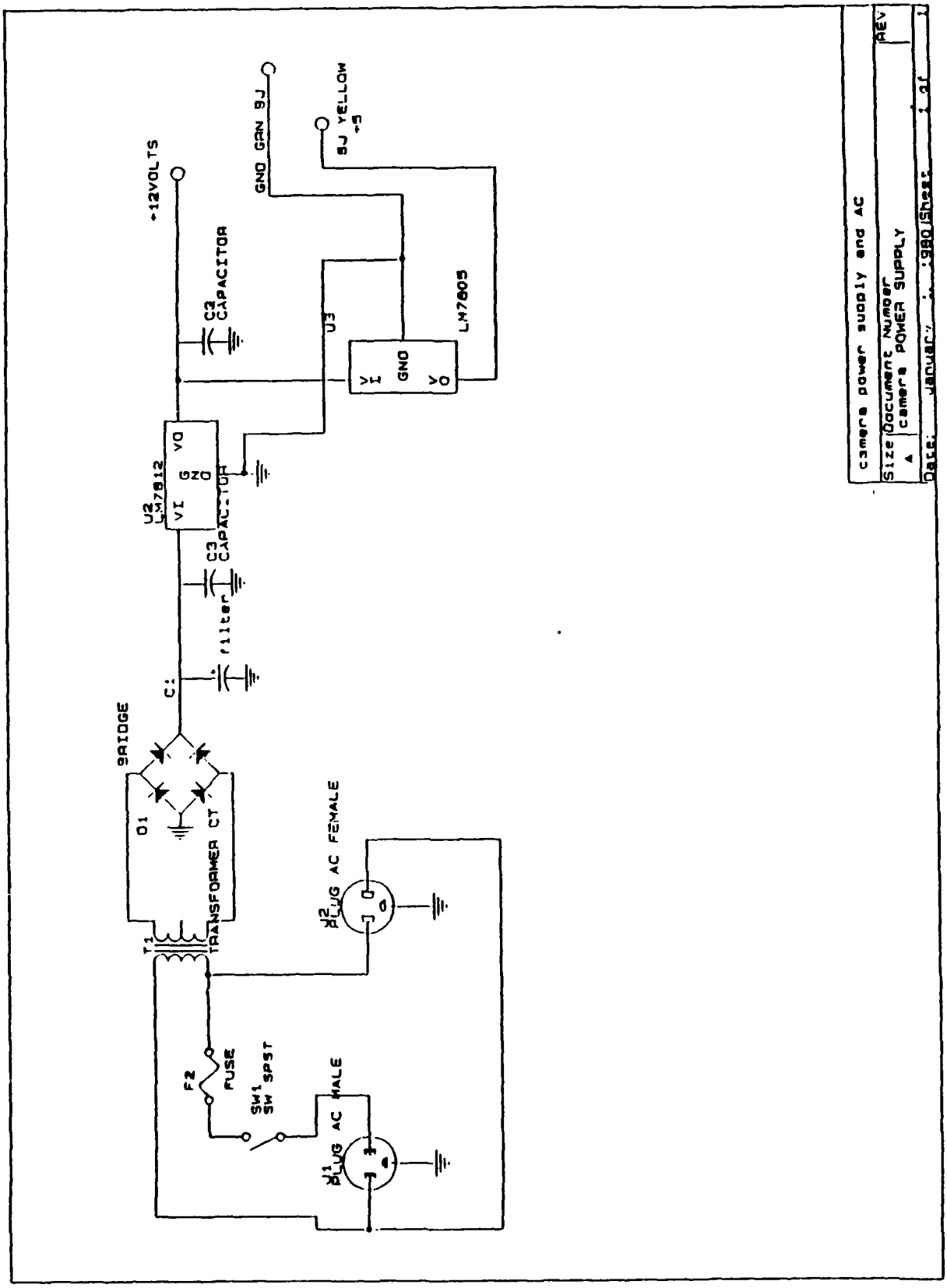

4 MHZ
CLOCK
VCC
C3
.1 uF
U1
X1
EX2
RESET
INT
TIMER
VPP
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
PB0
PB1
PB2
PB3
PB4
PB5
PB6
PB7
PC0
PC1
PC2
PC3
PC4
PC5
PC6
PC7
PD0
PD1
PD2
PD3
PD4
PD5
PD6/INT2
PD7
BLANK
MINUS
DECIMAL
FUNCTION
MODE
JP1 (P)
HEADER
CAPT. P. O. LLOYD USMC
Size
A
Document Number
CAMERA MPU (PAN)
REV
1
Date: December 22, 1989 Sheet: 1 of 1


CAPT. P. D. LLOYD USMC
Size: A
Document Number
CAMERA MPU OUTPUT (PAN)
REV 1
Date: December 22, 1988
Sheet 1 of 1
VCC
NC
ZC
APACK1 220
APACK2 220
APACK3 220
APACK4 220
APACK5 220
U7
U8
U9
U10
U11
R7 4.7K
SW3
LIGHT TEST
U6A
74LS32
MINUS
BLANK (P1)
DECIMAL (P1)
MINUS (P1)

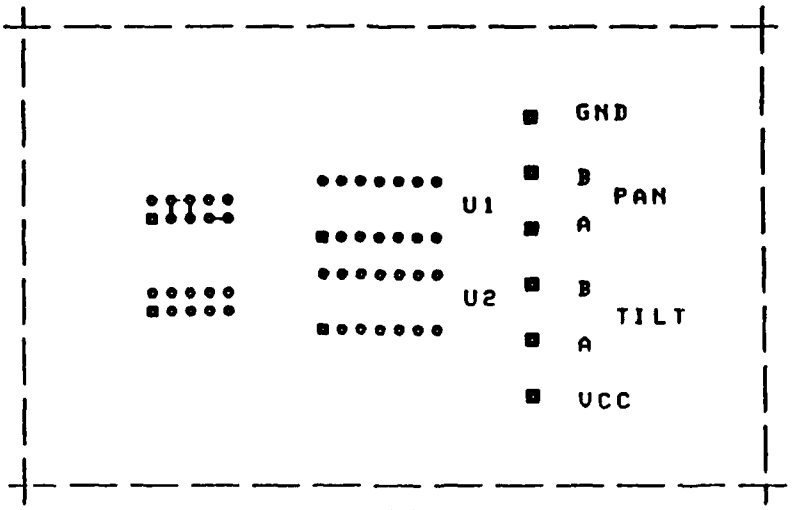

CAPT. P. D. LLOYD USMC
CAMERA MPU INPUT (TILT)
RESET (T)
INTERRUPT (T)
CH_A (T)
MODE (T)
CH_B (T)
FUNCTION (T)
SET (T)
VCC
R1 1 M
SW1 (T)
RESET
U2F 7414
U3A 74LS86
U2C 7414
C1 47 nF
U2B 7414
U2A 7414
J3 (T)
CH_A (T)
U2D 7414
J4 (T)
CH_B (T)
U4C 7414
U3D 74LS86
U4B 7414
C2 47 nF
U4A 7414
U5C 74S00
U5D 74S00
SET (HYSTERESIS) SW4 (T)
R6 1 K
R5 1 K
FUNCTION (HYSTERESIS) SW3 (T)
R4 4.7 K
U5B 74S00
U5A 74S00
R3 1 K
R2 1 K
DISPLAY MODE SW2 (T)


CAPT. P. D. LLOYD USMC
Size Document Number REV
A CAMERA MPU (TILT) 1
Date: December 22, 1989 Sheet 1 of
BLANK (T)
MINUS (T)
DECIMAL (T)
DIGIT 1 (T)
DIGIT 2 (T)
DIGIT 3 (T)
DIGIT 4 (T)
U1
X1
EX2
RESET
INT
TIMER
VPP
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
PB0
PB1
PB2
PB3
PB4
PB5
PB6
PB7
PC0
PC1
PC2
PC3
PC4
PC5
PC6
PC7
PD0
PD1
PD2
PD3
PD4
PD5
PD6/INT2
PD7
SAT05U3
VCC
4 MHZ CLOCK
C3
1 UF
JP1 (T)
HEADER
2 4 6 8 10 12 14
1 3 5 7 9 11 13
RESET (T)
INTERRUPT (T)
CH 1 (T)
MODE (T)
CH 2 (T)
FUNCTION (T)
SET (T)

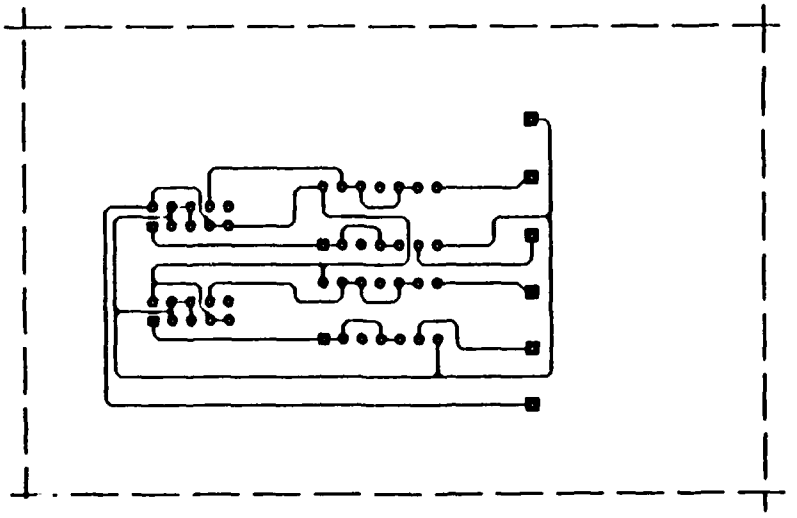

CAPT. P. D. LLOYD USMC
Size Document Number REV
A CAMERA MPU OUTPUT (TILT) 1
Date: December 22, 1989 Sheet 3 of 3
VCC
NC
RPACK1 220
RPACK2 220
RPACK3 220
RPACK4 220
U7
U8
U9
U11
74LS47
U8A
74LS32
A7 4.7K
SW5
LIGHT TEST
MINUS
BLANK
DECIMAL
MINUS
DIGIT 1
DIGIT 2
DIGIT 4
DIGIT 3


camera power supply and AC
Size: A
Document Number
camera POWER SUPPLY
REV
Date: January
Sheet
+12VOLTS
GND GRN 8J
8J YELLOW
+5
C2
CAPACITOR
U3
LM7805
VI
GND
VO
U2
LM7812
VI
GND
VO
C3
CAPACITOR
filter
C1
BRIDGE
D1
TRANSFORMER CT
T1
J2
PLUG AC FEMALE
F2
FUSE
SW1
SW SPST
J1
PLUG AC MALE

APPENDIX B. PRINTED CIRCUIT BOARD PLANS

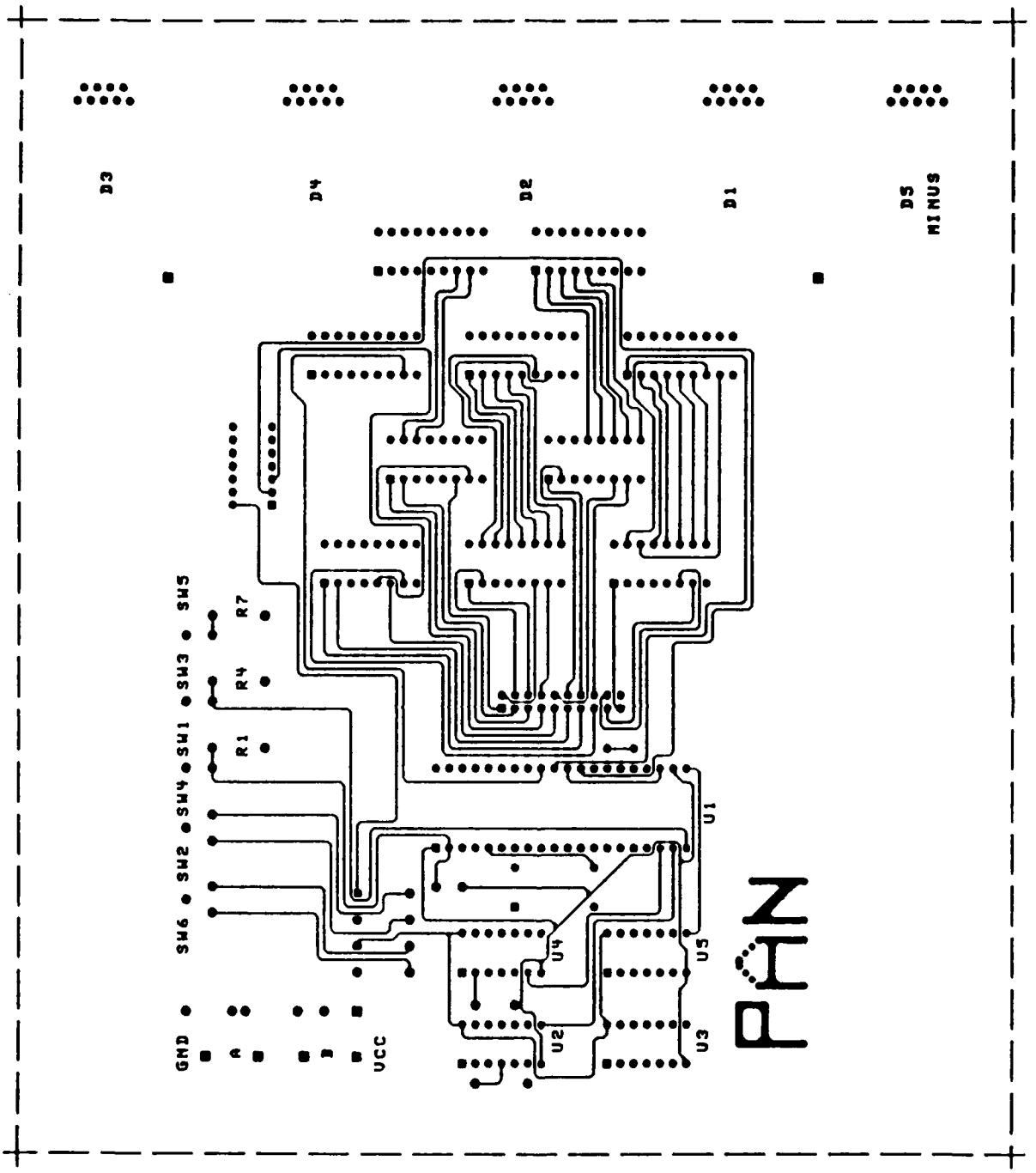


(a)

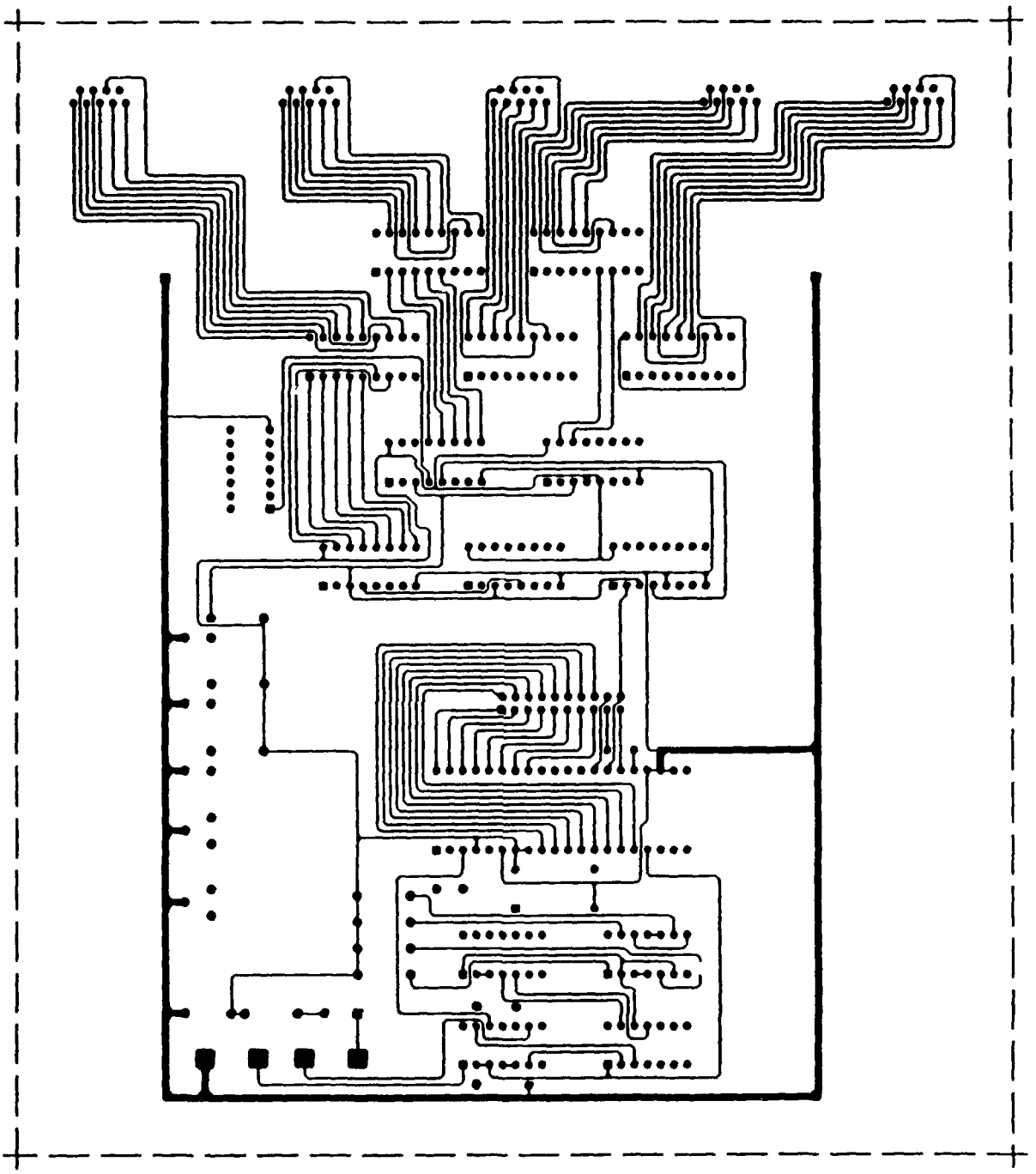


(b)

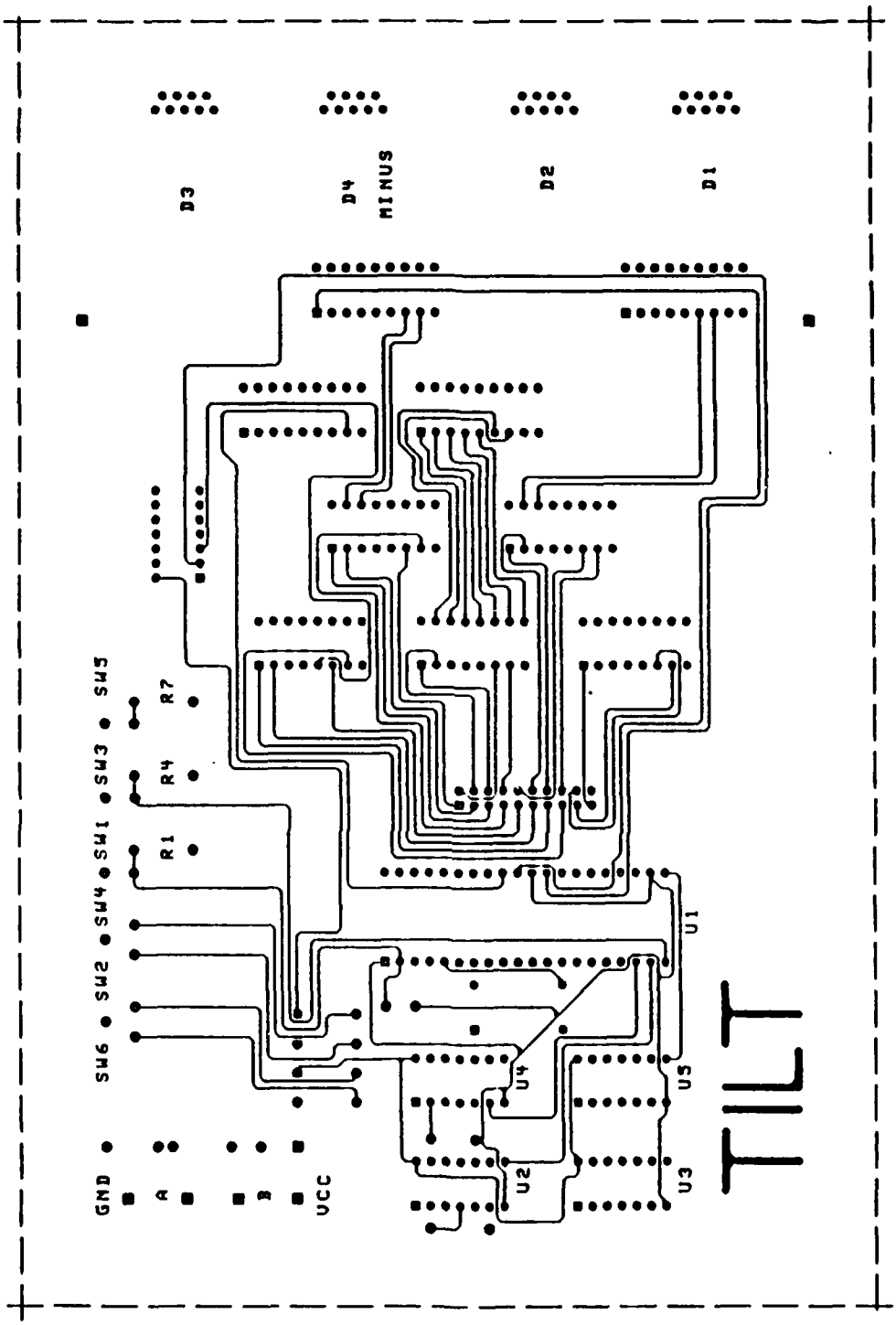
Line Driver, (a) Front and (b) Back. Full scale.



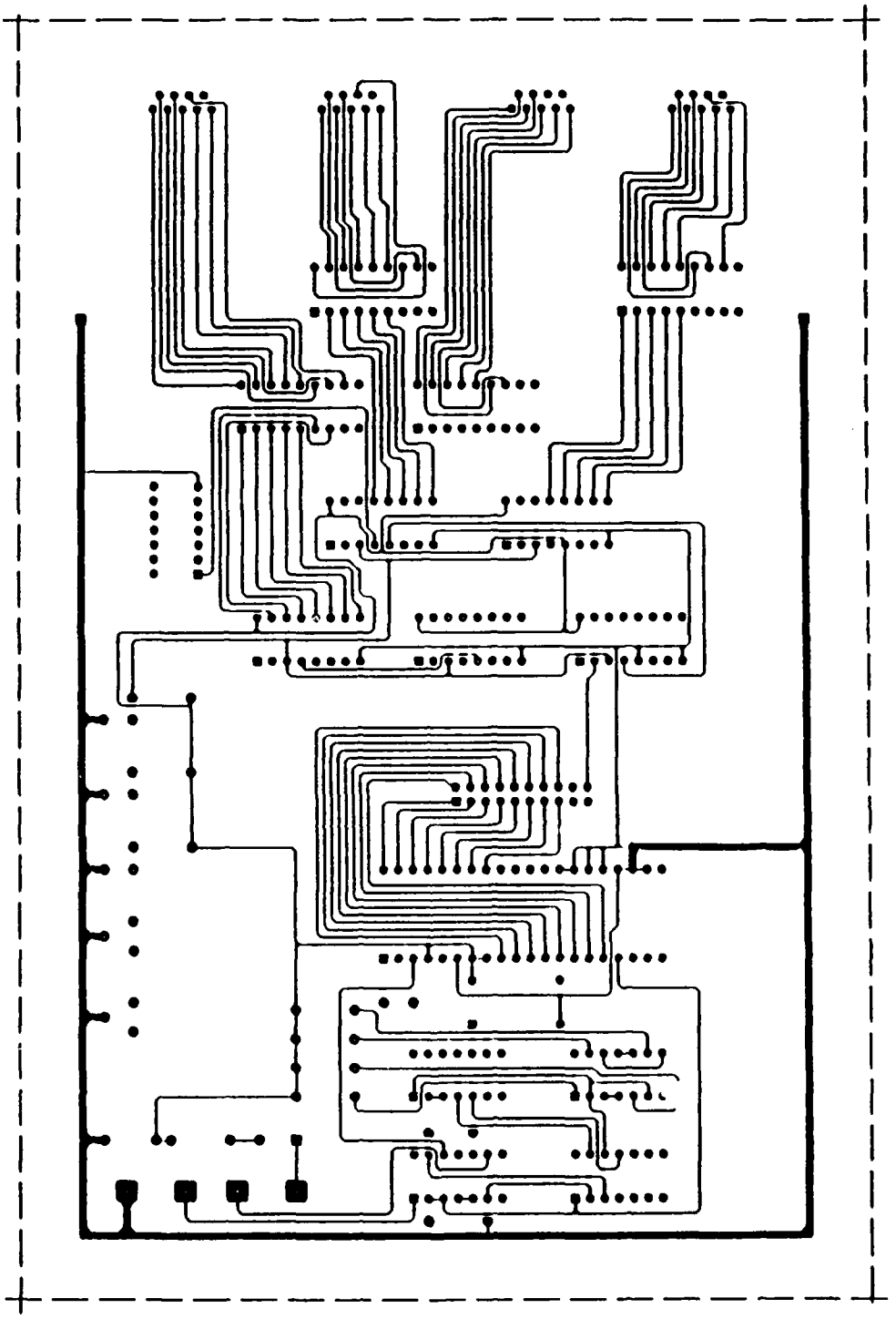
Pan MPU, Front. 90 % of full scale.



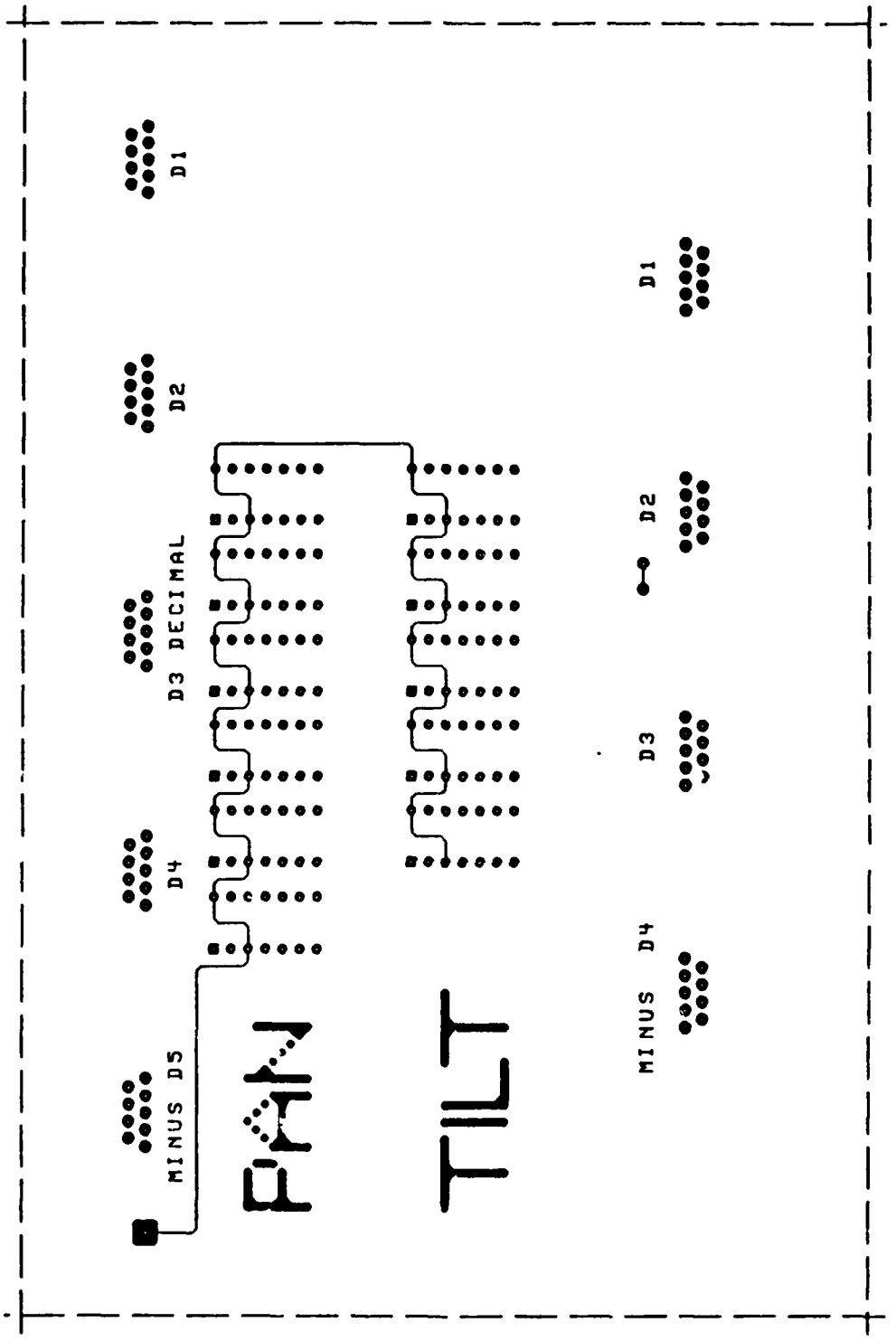
Pan MPU, Back. 90 % of full scale.



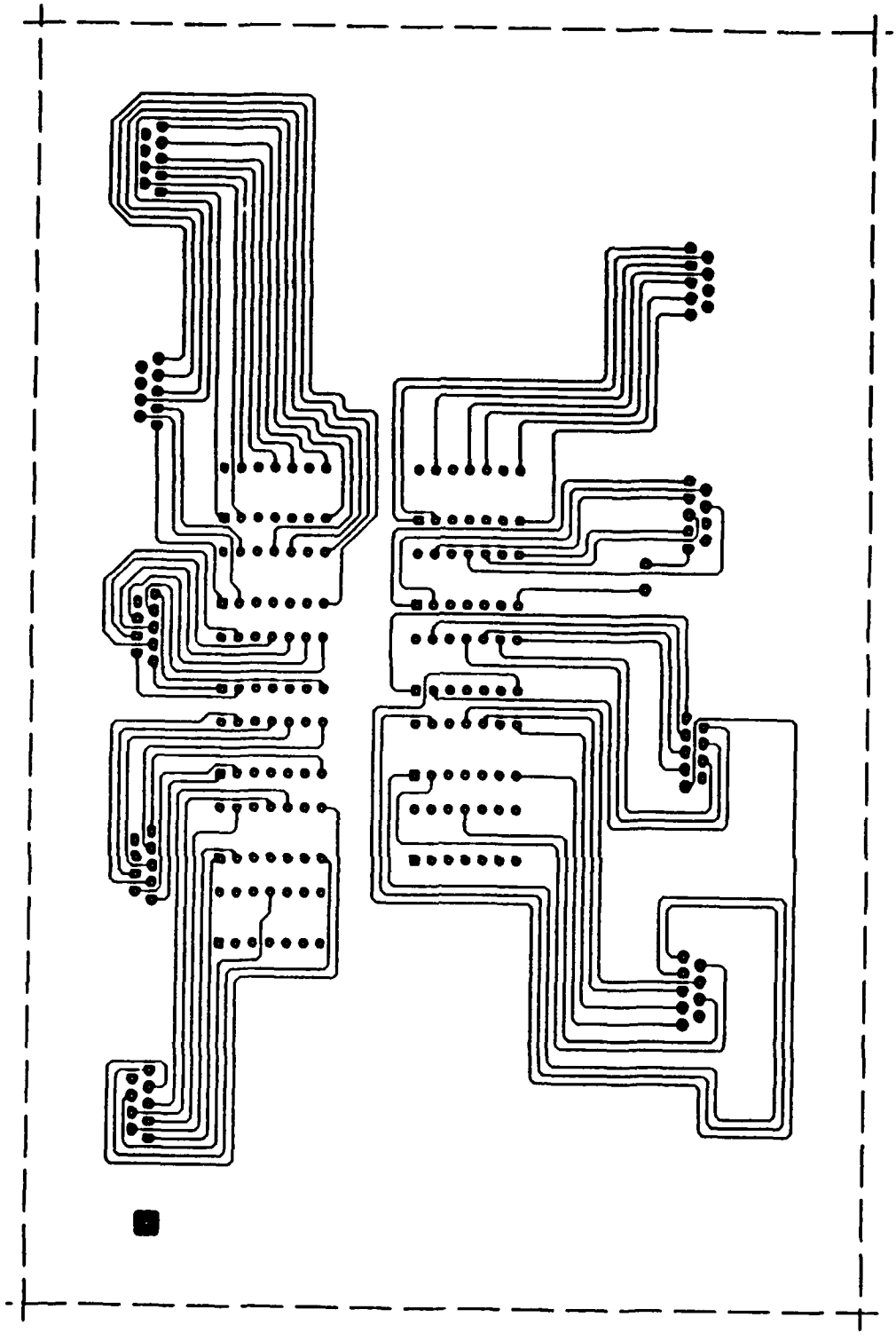
Tilt MPU, Front. 85 % of full scale.



Tilt MPU, Back. 85 % of full scale.



Display, Front. Full scale.



Display, Back. Full scale.

APPENDIX C. MICROPROCESSOR PROGRAMS

A. GENERAL

Sections B and C of this Appendix are the listing files for the two MPU programs written for the signal processor subsystems. The theory of operation of the two programs is identical and is most clearly described by Figure 31. The detailed operation of the Initialization , Mode Change, Blink, Output Display and Hysteresis Modify routines is described by the comments which accompany each of the programs. In addition to the detailed description provided by the program comments, the operation of the Count Routine is also explained in Chapter IV.

B. PAN

```
1                                TTL           POSITION DETERMINING PROGRAM (AZIMUTH)
2                       *                      LATEST REVISION              9 MAY 89
3                       *                      FILE NAME                    PAN.ASM
4                       **
5                       ** PROGRAM DESCRIPTION
6                       **
7                       **
8                       **
9                       ** I/O REGISTER ADDRESSES
10                      **
11      0000            PORTA   EQU     $0000   I/O PORT A
12      0001            PORTB   EQU     $0001   I/O PORT B
13      0002            PORTC   EQU     $0002   I/O PORT C
14      0003            PORTD   EQU     $0003   INPUT PORT D
15                      **
16                      ** DATA DIRECTION REGISTER OFFSET
17                      **
18      0004            DDR     EQU     4       (eg. DDR FOR PORT A IS  PORTA+DDR )
19                      **
20                      ** OTHERS
21                      **
22      0008            TIMER   EQU     $0008   EIGHT BIT TIMER REGISTER.
23      0009            TCR     EQU     $0009   TIMER CONTROL REGISTER.
24      000A            MR      EQU     $000A   MISCELLANEOUS REGISTER.
25      0010            RAM     EQU     $0010   START OF ON-CHIP RAM(112 - 31 FOR STACK)
26      0080            ZROM    EQU     $0080   PAGE ZERO OF ROM.
27      0100            ROM     EQU     $0100   START OF MAIN ROM.
28      0F38            MOR     EQU     $0F38   MASK OPTION REGISTER.
29      0FF8            INTRPT  EQU     $0FF8   LOCATION OF INTERRUPT VECTORS.
30      1000            MEMSIZ  EQU     $1000   MEMORY ADDRESS SIZE.
31                      **
32                      ** EQUATES
33                      **
34      0001            BIT0    EQU     1
35      0002            BIT1    EQU     2
36      0004            BIT2    EQU     4
37      0008            BIT3    EQU     8
38      0010            BIT4    EQU     16
39      0020            BIT5    EQU     32
40      0040            BIT6    EQU     64
41      0080            BIT7    EQU     128
42                      **
43      0000            B0      EQU     0
44      0001            B1      EQU     1
45      0002            B2      EQU     2
46      0003            B3      EQU     3
47      0004            B4      EQU     4
48      0005            B5      EQU     5
49      0006            B6      EQU     6
50      0007            B7      EQU     7
51                      **
52                      ** EQUATES FOR THE TIMER CONTROL REGISTER
53                      **
54                      ***
55      0007            TIR     EQU     7       TIMER INTERRUPT REQUEST. 1 = REQUEST, 0 = NO REQ.
56      0006            TIM     EQU     6       TIMER INTERRUPT MASK. 1 = DISABLED, 0 = ENABLED.
57      0005            TIN     EQU     5       EXTERNAL OR INTERNAL CLOCK SOURCE. 1 = EXT, 0 = INT.
58      0004            TEE     EQU     4       EXTERNAL CLOCK ENABLE. NOT USED.
59      0003            PSC     EQU     3       PRESCALER CLEAR. NOT USED.
60      0002            PS2     EQU     2       (PS2)  --
61      0001            PS1     EQU     1       (PS1)    |--  PRESCALER SELECT BITS.
```

```
0000            PS0     EQU     0       (PS0) --
                **
                ** EQUATES FOR THE STATUS BYTE, 'STAT'.
                ***
                ***
                **
0007            UD      EQU     7       COUNT DIRECTION?  1 = UP, 0 = DOWN.
0006            MOD_32  EQU     6       IS 'BINCT' MODULO 32?   1 = YES, 0 = NO.
0005            FLASH   EQU     5       BLINK THE DISPLAY?  1 = YES, 0 = NO.
0004            POSCT   EQU     4       DISPLAY POSITION OR COUNT?  1 = POS, 0 = COUNT.
0003            L_SET   EQU     3       VALUE OF 'MODE,PORTD' LAST TIME.
0002            NEGTIV  EQU     2       IS 'BCDCT'NEGATIVE NUMBER?  1 = YES, 0 = NO.
                ***             1       NOT USED.
                ***             0       NOT USED.
                ***
                ***
                ** I/O EQUATES AND DESCRIPTIONS.
                ***
                ***
                ***      PORT A  (I/O)
                ***
                ***     +-------+-------+-------+-------+-------+-------+-------+-------+
                ***     |          BCD DIGIT #4         |          BCD DIGIT #3         |
                ***     +-------+-------+-------+-------+-------+-------+-------+-------+
                ***     |  D4   |  C4   |  B4   |  A4   |  D3   |  C3   |  B3   |  A3   |
                ***     +-------+-------+-------+-------+-------+-------+-------+-------+
                *** BIT     7       6       5       4       3       2       1       0
                ***
                ***
                ***      PORT B  (I/O)
                ***
                ***     +-------+-------+-------+-------+-------+-------+-------+-------+
                ***     |        DISPLAY CONTROL .      |BCD DIGIT #5 (MOST SIGNIFICANT)|
                ***     +-------+-------+-------+-------+-------+-------+-------+-------+
                ***     | DECPT | POSTIV|       | BLANK |   D5  |   C5  |   B5  |   A5  |
                ***     +-------+-------+-------+-------+-------+-------+-------+-------+
                *** BIT     7       6       5       4       3       2       1       0
                ***
0007            DECPT   EQU     7       TO DISPLAY THE DECIMAL POINT...DECPT IS CLEARED
0006            POSTIV  EQU     6       USED TO DISPLAY NEGATIVE SIGN...CLEARED TO SHOW
                ***                     MINUS SIGN.
0004            BLANK   EQU     4       TO BLANK DIGITS 2 AND 3...CLEAR BLANK.
                ***                     DIGITS 4 AND 5 ARE ALWAYS BLANKED.
                ***                     DIGIT 1 IS NEVER BLANKED.
                ***
                ***      PORT C  (I/O)
                ***
                ***     +-------+-------+-------+-------+-------+-------+-------+-------+
                ***     |          BCD DIGIT #2         |BCD DIGIT #1(LEAST SIGNIFICANT)|
                ***     +-------+-------+-------+-------+-------+-------+-------+-------+
                ***     |  D2   |  C2   |  B2   |  A2   |  D1   |  C1   |  B1   |  A1   |
                ***     +-------+-------+-------+-------+-------+-------+-------+-------+
                *** BIT     7       6       5       4       3       2       1       0
                ***
                ***
                ***      PORT D  (INPUT ONLY)
                ***
                ***     +-------+-------+-------+-------+-------+-------+-------+-------+
                ***     | CH_A  | INT2  | CH_B  | FUNCT |  SET  |       |       |       |
                ***     +-------+-------+-------+-------+-------+-------+-------+-------+
                *** BIT     7       6       5       4       3       2       1       0
                ***
0007            CH_A    EQU     7       INDICATES THE STATUS OF CHANNEL A.
0006            INT2    EQU     6       INTERRUPT #2.  USED TO CHANGE DISPLAY MODES.
0005            CH_B    EQU     5       INDICATES THE STATUS OF CHANNEL B.
0004            FUNCT   EQU     4       USED TO PUT THE PROGRAM IN A MODE THAT WILL ALLOW
```

```
                        ***                     'HYST' TO BE INCREMENTED.
        0003            SET     EQU     3       INCREMENTS 'HYST' WHEN TOGGLED AND FUNCT IS LOW.
                        ***
                        ***
                        ********************************************************************************
                        **                                                                              *
                        **                          RAM VARIABLES                                       *
                        **                                                                              *
                        ********************************************************************************
                        **
                        ** RESERVE MEMORY SPACE FOR THE PROGRAM VARIABLES.
                        **
0000                            DATA
                        **
                        **
0000                            ABSOLUTE (ABSOLUTE ADDRESSING USED HERE TO RELATIVE DIRECTIVE)
                        **
0010                            ORG     RAM     START OF RAM.
                        **
                        *** BINARY COUNT.
0010                    BINCT   RMB     3
        0010            HIBIN   EQU     BINCT   HI BYTE.
        0011            MIDBIN  EQU     BINCT+1 MIDDLE BYTE.
        0012            LOBIN   EQU     BINCT+2 LO BYTE.
                        **
                        *** POSITION POINTERS.
0013                    PTR     RMB     4       EACH BYTE POINTS TO A POSITION IN THE
                        **                      TABLE THAT CONTAINS ONE OR TWO DIGITS
                        **                      OF THE BCD POSITION.
        0013            PTR4    EQU       PTR   MOST SIGNIFICANT DIGITS.
        0014            PTR3    EQU       PTR+1
        0015            PTR2    EQU       PTR+2
        0016            PTR1    EQU       PTR+3 LEAST SIGNIFICANT DIGIT.
                        **
                        *** COUNT POINTERS.
0017                    CTPTR   RMB     3       EACH BYTE POINTS TO A POSITION IN THE
                        **                      TABLE THAT CONTAINS TWO OF THE DIGITS
                        **                      IN THE BCD COUNT.
        0017            CTPTR3  EQU     CTPTR   MOST SIGNIFICANT DIGITS.
        0018            CTPTR2  EQU     CTPTR+1
        0019            CTPTR1  EQU     CTPTR+2 LEAST SIGNIFICANT DIGITS.
                        **
                        *** BCD POSITION IN DEGREES.
001A                    DEGRES  RMB     4
        001A            HUNDEG  EQU     DEGRES      CONTENTS X 100.000
        001B            ONEDEG  EQU     DEGRES+1    CONTENTS X   1.000
        001C            HUNDTH  EQU     DEGRES+2    CONTENTS X   0.010
        001D            THOUTH  EQU     DEGRES+3  + CONTENTS X   0.001
                        ***                       ---------------------
                        ***                         POSITION IN DEGREES
                        **
                        *** BCD COUNT.
001E                    BCDCT   RMB     3
        001E            TENTHO  EQU     BCDCT       CONTENTS X 10,000
        001F            HUNDRD  EQU     BCDCT+1     CONTENTS X    100
        0020            TENONE  EQU     BCDCT+2   + CONTENTS X      1
                        ***                       ----------------------
                        ***                       NUMBER OF PULSES COUNTED
                        ***
                        *** HYSTERESIS COUNTER. POINTS TO A NUMBER IN THE TABLE THAT IS THE
                        ***      AMOUNT OF HYSTERESIS PRESENT IN THE SYSTEM.  INITIALIZED TO 7.
0021                    HYSTPT  RMB     1
                        **
                        *** POSITION INCREMENT.  CONTAINS A NUMBER, THAT WHEN MULTIPLIED BY 0.001
                        ***      IS THE NUMBER OF DEGREES THAT THE POSITION COUNTER (BCDPOS) IS
                        ***      TO BE INCREMENTED OR DECREMENTED DURING PROGRAM EXECUTION.
```

```
                        ***     THE VALUE OF 'POSINC', DETERMINED EXPERIMENTALLY, SHOULD BE
                        ***     7.0312.  SINCE THE PROGRAM IS DESIGNED WORK WITH INTEGERS ONLY
                        ***     THIS NUMBER IS ROUNDED TO 7.  TO REDUCE THE CUMULATIVE EFFECT OF
                        ***     THE ROUND OFF, EVERY 32 COUNTS 'POSINC' IS SET EQUAL TO 8.  THIS
                        ***     AGAIN LEADS TO SOME CUMULATIVE ERROR THAT IS ACCOUNTED FOR BY
                        ***     SETTING 'POSINC' TO 7 INSTEAD OF 8 WHEN THE COUNT REACHES A
                        ***     VALUE OF 16384 (2^14).
                        ***
0022                    POSINC  RMB     1
                        ***
                        *** HYSTERESIS VARIABLES.  USED TO ELIMINATE THE EFFECTS OF BACKLASH ON
                        ***     THE POSITION MEASUREMENTS.
                        ***
0023                    HYST    RMB     1       THE THRESHHOLD VALUE DETERMINED
                        ***                     EXPERIMENTALLY.
0024                    HYSTCT  RMB     1       CURRENT AMOUNT OF HYSTERESIS MEASURED.
                        ***
                        *** STATUS BYTE.  USED TO KEEP TRACK OF WHAT IS GOING ON.
                        ***
0025                    STAT    RMB     1       CURRENT STATUS.
0026                    LSTAT   RMB     1       PREVIOUS/LAST STATUS.  USED TO KEEP TRACK OF
                        ***                     L_SET ONLY.
                        ***
                        *** TIMER COUNTER.  USED IN CONJUNCTION WITH THE TIMER PRESCALER AND THE
                        ***     TDR TO KEEP TRACK OF ONE SEC. INTERVALS.  USED IN BLINKING THE
                        ***     DISPLAY.  INITIALLY SET TO 31, WHEN THE 'FLASH' BIT OF 'STAT'
                        ***     IS SET.  TIMCT IS DECREMENTED EACH CLOCK INTERRUPT (APPROX. 31
                        ***     TIMES PER SEC).  RESET TO 31 WHEN CONTENTS GO TO ZERO.
                        ***     WHEN (TIMCT)=0 THE DISPLAY WILL TOGGLE.
                        ***
0027                    TIMCT   RMB     1
                        **
                                ENDS
                        **
                        ***
                        ********************************************************************
                        **                                                                  *
                        **                      PAGE ZERO ROM
                        **                                                                  *
                        ********************************************************************
                        **
                        **                  INITIALIZATION ROUTINE.
                        **
                        **
0000                            CODE
                        **
0080                            ORG     ZROM    PAGE ZERO ROM.
                        **
0080                            RELATIVE        RELATIVE ADDRESSING MUST BE USED FOR THE BRANCH.
                        **
        0080            RESTRT  EQU     $       THIS IS THE ENTRY POINT WHEN THE RESET
                        ***                     SWITCH IS PUSHED.
                        **
                        ********************************************************************
                        **
                        ***             INITIALIZE THE PC AND CLEAR RAM.
                        ***
                        ***
0080    98                      SEI             SET INTERUPT TO AVOID INTERUPTION AND
0081    9C                      RSP             RESET THE STACK POINTER. JUST IN CASE!
                        ***
0082    AE 10                   LDX     #BINCT  CLEAR ALL OF THE VARIABLES BETWEEN
0084    7F              CLRIT   CLR     ,X      'BINCT' AND 'TIMCT' (INCLUSIVE).NOTE
0085    5C                      INCX            THAT THIS SETS THE COUNTER AND THE POS-
0086    A3 27                   CPX     #TIMCT  ITION TO ZERO. THIS MEANS THAT ROTATION
0088    23 FA                   BLS     CLRIT   SHOULD START IN AN INCREASING (CW)
```

```
                        ***                     DIRECTION FROM THE MOST CCW POSITION
                        ***                     AFTER A RESET.
                        ***
008A                            ABSOLUTE        BACK TO ABSOLUTE ADDRESSING.
                        ***
                        ********************************************************************************
                        ***
                        ***                         ESTABLISH I/O PORTS.
                        ***
008A  A6 FF                     LDA     #-1         PORTS A,B,C ARE CONFIGURED AS
008C  B7 04                     STA     PORTA+DDR   ALL OUTPUT. PORT D IS ALL INPUT
008E  B7 05                     STA     PORTB+DDR   SO THERE IS NO MASK TO SET.
0090  B7 06                     STA     PORTC+DDR
                        ***
                        ***
0092  CD 03 4E                  JSR     OUTCT       COUNT IS TO BE DISPLAYED INITIALLY.
                        ***
                        ********************************************************************************
                        ***                 SET UP THE STATUS REGISTER.
                        ***
0095  A6 08                     LDA     #%00001000  --
0097  B4 03                     AND     PORTD         |--> SET UP 'L_SET' BIT OF 'STAT'.
0099  B7 25                     STA     STAT        --
                        ***
009B  1C 25                     BSET    MOD_32,STAT  0 IS MODULO 32.
                        **
                        ********************************************************************************
                        ***
                        ***                     INITIALIZE HYSTCT.
                        ***
009D  A6 08                     LDA     #08
009F  B7 23                     STA     HYST
00A1  B7 21                     STA     HYSTPT
                        ***
                        ********************************************************************************
                        **
                        **        SET UP THE TIMER FOR A 4 MHZ CRYSTAL / 4 = 1 MHZ CLOCK.
                        ***
                        ***    NOTE:  THE MASK OPTION REGISTER IS IN ROM.  IT IS SET UP AT
                        ***           THE END OF THE PROGRAM.
                        ***
                        **********************   SET UP THE TCR.  **********************************
                        ***
00A3  A6 47                     LDA       #BIT6+BIT2+BIT1+BIT0
                        ***             (TIM)|(PS2)(PS1)(PS0)
                        ***(DISABLE INTERRUPT)|(PRESCALE BY 128)
                        ***
00A5  B7 09                     STA     TCR
                        ***
                        ***
                        *** SET UP THE TIMER.
                        ***
00A7  A6 FF                     LDA     #255    1 MHZ/(128*255) = 30.6 (APPROX. 31)
00A9  B7 08                     STA TIMER
                        ***
                        ****************  INITIALIZE THE TIMER COUNTER.  ************************
                        ***
00AB  A6 1F                     LDA     #31     PROVIDES FOR 1 SEC. BLINK INTERVAL.
00AD  B7 27                     STA     TINCT   FOR 2 SEC. INTERVAL JUST USE TIMECT=62, etc.
                        ***
                        ***
                        ********************************************************************************
                        **
                        **                      SET UP MISCELLANEOUS REGISTER.
                        **
00AF  1D 0A                   BCLR    B6,MR     ENABLES THE SECOND INTERRUPT.
```

```
                        **
                        **
00B1  9A                        CLI             CLEAR THE INTERRUPT MASK TO GET STARTED.
                        ***
00B2                            RELATIVE        RELATIVE ADDRESSING MUST BE USED FOR THE
                        ***                     REMAINDER OF THE PROGRAM.
                        ***
                        ********************************************************************************
                        **
                        ** WAIT LOOP.  EXECUTES, UNTIL AN INTERRUPT OCCURS.
                        **
00B2  09 03 08          PAUSE   BRCLR   FUNCT,PORTD,CHHYST WANT TO CHANGE HYST?
                        **                              YES...GO TO CHHYST.
00B5  06 03 04                  BRSET   SET,PORTD,SBIT  NO...'SET,PORTD' SET?
00B8  17 25                     BCLR    L_SET,STAT          NO...CLEAR 'L_SET,STAT'
00BA  20 F6                     BRA     PAUSE               AND LOOP.
00BC  16 25             SBIT    BSET    L_SET,STAT          YES...SET 'L_SET,STAT'
00BE  20 F2                     BRA     PAUSE               AND...LOOP
                        **
                        ********************************************************************************
                        **
                        ** HYSTERESIS MODIFICATION ROUTINE.  PERMITS MODIFICATION OF THE
                        ** HYSTERESIS BUFFER WITHOUT REPROGRAMMING.
                        **
00C0  A6 40             CHHYST  LDA     #BIT6           DISABLE TIMER INTERRUPT.
00C2  B7 09                     STA     TCR
                        ***
00C4  A6 08                     LDA     #%00001000      SAVE 'L_SET'
00C6  B4 25                     AND     STAT            INTO
00C8  B7 26                     STA     LSTAT           'LSTAT'.
00CA  A6 08                     LDA     #%00001000
00CC  B4 03                     AND     PORTD           'SET,PORTD' --> ACCUMULATOR
00CE  B1 26                     CMP     LSTAT           HAS THE SET SWITCH BEEN CHANGED?
00D0  27 15                     BEQ     DISPLA
                        ***                             YES...
00D2  A6 08                     LDA     #%00001000      --
00D4  B8 25                     EOR     STAT              |-->CHANGE 'L_SET,STAT',
00D6  B7 25                     STA     STAT            --
00D8  3C 23                     INC     HYST            THEN INCREMENT THE HYSTERESIS
00DA  3C 21                     INC     HYSTPT          POINTER AND 'HYST'...
                        ***
00DC  B6 21                     LDA     HYSTPT        --
00DE  A1 19                     CMP     #25             --
00E0  23 05                     BLS     DISPLA            --
                        ***                                 -- BUT NOT ABOVE 25.
00E2  4F                        CLRA                      --      THEN--->|
00E3  B7 23                     STA     HYST            --                |
00E5  B7 21                     STA     HYSTPT        --                  |
                        ***                                               |
00E7  A6 00             DISPLA  LDA     #%00000000      NO... JUST--------->|
00E9  B7 00                     STA     PORTA         --                  |
00EB  A6 C0                     LDA     #%11000000      --                |
00ED  B7 01                     STA     PORTB             --             \|/
                        ***                                   -- DISPLAY CURRENT 'HYST'.
00EF  BE 21                     LDX     HYSTPT            --
00F1  D6 03 BE                  LDA     TABLE,X         --
00F4  B7 02                     STA     PORTC         --
                        ***                             IS 'HYST' SETTING COMPLETE?
00F6  09 03 C7                  BRCLR   FUNCT,PORTD,CHHYST NO... KEEP CHECKING 'SET'.
00F9  08 25 05                  BRSET   POSCT,STAT,SHOPOS  YES... RESET THE DISPLAY.
00FC  CD 03 4E                  JSR     OUTCT
00FF  20 03                     BRA     DUNCHG
                        ***
0101  CD 03 68          SHOPOS  JSR     OUTPOS
                        ***
0104  0B 25 04          DUNCHG  BRCLR   FLASH,STAT,NO_INT  IF THE DISPLAY IS TO BLINK...
```

```
0107  A6 07          LDA   #BIT2+BIT1+BIT0   ENABLE TIMER INTERRUPT AND RESET
0109  B7 09          STA   TCR               TIMER PRESCALER
010B  20 A5    NO_INT  BRA   PAUSE           PRIOR TO RETURNING.
               ***
               **
               ********************************************************************************
               ** MAXIMUM EXECUTION TIME FOR THE REMAINDER OF THE PROGRAM OCCURS
               ** IF THE COUNTER ROTATES THROUGH ZERO AS THE DISPLAY MODE IS CHANGED
               ** FROM THE BLINKING MODE TO THE COUNT MODE AT THE SAME TIME THAT THE
               ** BLINKING ROUTINE IS CAUSING THE DISPLAY TO TOGGLE TO SHOW THE
               ** POSITION IN DEGREES.
               **     MAXIMUM EXECUTION TIME = 140 + 184 + 708 = 1032 CLOCK CYCLES.
               ********************************************************************************
               **
               ** MODE CHANGE ROUTINE.  CHANGES THE DISPLAY MODE FROM
               **       COUNT -> POSITION -> BLINKING -> COUNT ->....(ETC.)
               **       MAXIMUM EXECUTION TIME OF 181 CLOCK CYCLES OCCURS WHEN THE
               **       DISPLAY MODE IS CHANGED FROM DISPLAYING THE COUNT TO DISPLAYING
               **       THE POSITION (IN DEGREES).
               **       IF THE DISPLAY IS CHANGED FROM BLINKING TO A COUNT DISPLAY
               **       EXECUTION TIME IS 140 CLOCK CYCLES.
               **
010D  1F 0A    CHMODE  BCLR  B7,MR             AVOID REPEATED INTERRUPTS.
               **
010F  0A 25 10         BRSET FLASH,STAT,DIS_CT  IF FLASHING, DISPLAY COUNT...
0112  09 25 07         BRCLR POSCT,STAT,DISPOS  IF SHOWING COUNT, DISPLAY POSITION...
0115  1A 25            BSET  FLASH,STAT         ELSE, BLINK.
               **
0117  A6 07           LDA   #BIT2+BIT1+BIT0    ENABLE TIMER INTERRUPT AND RESET
0119  B7 09           STA   TCR                TIMER PRESCALER.
011B  80               RTI
               **
011C  18 25    DISPOS  BSET  POSCT,STAT      --
011E  CD 03 68         JSR   OUTPOS            |-- DISPLAY CURRENT POSITION, AND WAIT.
0121  80               RTI                   --
               **
0122  A6 47    DIS_CT  LDA   #BIT6+BIT2+BIT1+BIT0  DISABLE TIMER INTERRUPT AND RESET
0124  B7 09           STA   TCR                    TIMER PRESCALER.
0126  19 25           BCLR  POSCT,STAT      --
0128  1B 25           BCLR  FLASH,STAT        |
               **                             |-- DISPLAY CURRENT COUNT, AND WAIT.
012A  CD 03 4E         JSR   OUTCT             |
012D  80               RTI                   --
               **
               ********************************************************************************
               **
               ** BLINK ROUTINE.  INTERRUPT ROUTINE TO CHANGE THE DISPLAY FROM POSITION
               **       TO COUNT OR VICE VERSA EVERY 31 ST TIMER INTERRUPT IF THE
               **       'FLASH' BIT OF 'STAT' IS SET.
               **       MAXIMUM EXECUTION TIME OF 184 CLOCK CYCLES OCCURS WHEN THE
               **       DISPLAY IS TOGGLED FROM A COUNT DISPLAY TO A POSITION DISPLAY.
               **
012E           BLINK   EQU   $
012E  0F 09 DC         BRCLR TIR,TCR,CHMODE       IF THE INTERRUPT WAS NOT A TIMER
               **                                 INTERRUPT IT MUST BE FROM INT2.
               **
0131  1F 09            BCLR  TIR,TCR              AVOID REPEATED TIMER INTERRUPTS.
               **
0133  3A 27           DEC   TIMCT                IF THERE HAVE BEEN 31 TIMER
0135  27 01           BEQ   CHGDIS               INTERRUPTS  (1 SEC), IT'S TIME TO
               **                                 CHANGE THE DISPLAY.
0137  80               RTI                        OTHERWISE, IT'S BACK TO WORK.
               **
0138  A6 1F    CHGDIS  LDA   #31                  RESET TIMCT TO 31 (1 SEC. INTERVAL).
013A  B7 27           STA   TIMCT
               **
```

```
458  013C  B6 25          LDA     STAT            --
459  013E  A8 10          EOR     #%00010000        |-- CHANGE 'POSCT' BIT OF 'STAT'.
460  0140  B7 25          STA     STAT            --
461                  **
462  0142  08 25 04       BRSET   POSCT,STAT,POSOUT     DECIDE ON CORRECT DISPLAY.
463                  **
464                  ************ CHANGE THE DISPLAY TO SHOW THE COUNT..... ***************
465                  **
466  0145  CD 03 4E       JSR     OUTCT
467  0148  80             RTI
468                  **
469                  ************ OR HAVE THE DISPLAY SHOW THE POSITION. ******************
470                  **
471  0149  CD 03 68  POSOUT JSR     OUTPOS
472  014C  80             RTI
473                  **
474                  ********************************************************************************
475                  **
476                  **                          COUNT ROUTINE.
477                  **        WHEN A COUNT IS RECEIVED THIS IS THE ENTRY POINT .
478                  **        MAXIMUM EXECUTION TIME OF 708 CLOCK CYCLES OCCURS WHEN THE
479                  **        COUNTER ROTATES CCW THROUGH ZERO AND THE POSITION (IN DEGREES)
480                  **        IS BEING DISPLAYED.
481                  **
482                  **
483                  ** CURRENT DIRECTION OF ROTATION IS DETERMINED BY INSPECTING THE STATUS
484                  ** OF 'CH_A' AND 'CH_B'.  THE FOUR POSSIBILITIES AND THE ASSOCIATED
485                  ** DIRECTION OF ROTATION ARE AS SHOWN BELOW.  NOTE THAT THIS SCHEME
486                  ** PREVENTS MULTIPLE OSCILLATIONS ABOUT A SINGLE POINT FROM
487                  ** REPEATEDLY INCREMENTING OR DECREMENTING THE COUNTER.
488                  **
489                  **   +--------+--------+-----------------+------------------+
490                  **   |  CH_A  |  CH_B  |    DIRECTION    |  COUNT THE PULSE? |
491                  **   |        |        |  OF ROTATION    |                   |
492                  **   +--------+--------+-----------------+------------------+
493                  **   |   LO   |   LO   |       CW        |        NO         |
494                  **   |   LO   |   HI   |       CCW       |        YES        |
495                  **   |   HI   |   LO   |       CCW       |        NO         |
496                  **   |   HI   |   HI   |       CW        |        YES        |
497                  **   +--------+--------+-----------------+------------------+
498                  **
499                  ********  FIRST SEE IF WE ARE SUPPOSED TO COUNT THIS INTERUPT.  ********
500                  **
501       014D       COUNT   EQU     $
502  014D  0A 03 01       BRSET   CH_B,PORTD,OKCT     IF CH_B IS LO WE DON'T COUNT THE
503  0150  80             RTI                         INTERRUPT.
504                  **
505                  ************** IF THE INTERRUPT IS VALID UPDATE 'STAT'.  **************
506                  **
507  0151  A6 7F     OKCT    LDA     #%01111111
508  0153  B4 25          AND     STAT            SAVE ALL OF THE OLD 'STAT' EXCEPT THE
509  0155  B7 25          STA     STAT            DIRECTION OF ROTATION.
510  0157  A6 80          LDA     #%10000000
511  0159  B4 03          AND     PORTD           'CH_A,PORTD' INDICATES THE DIRECTION
512                  *                            OF ROTATION AND BECOMES 'UD,STAT'.
513  015B  BA 25          ORA     STAT            ADD THE RESULTS TO GET
514  015D  B7 25          STA     STAT            THE NEW 'STAT'.
515                  **
516                  ** DECIDE IF THE "SLACK" DUE TO BACKLASH/HYSTERESIS HAS BEEN TAKEN OUT.
517                  **
518  015F  0E 25 09       BRSET   UD,STAT,HYSTCK  IF ROTATING CW SEE BELOW.
519  0162  B6 24          LDA     HYSTCT          ELSE, SEE IF WE DECREMENT THIS TIME.
520  0164  27 60          BEQ     CCW             IF HYSTCT=0, GO TO THE COUNT DOWN
521                  **                           ROUTINE.
522  0166  A0 01          SUB     #1              ELSE, DECREMENT THE HYSTERESIS COUNTER,
523  0168  B7 24          STA     HYSTCT
```

```
524  016A  80              RTI                     AND WAIT FOR THE NEXT INTERRUPT.
525                    **
526  0168  B6 23     HYSTCK LDA     HYST            IF ROTATING CW....
527  016D  B1 24            CMP     HYSTCT          AND  HYST = HYSTCT ....
528  016F  27 03            BEQ     CW              COUNT THE PULSE .
529  0171  3C 24            INC     HYSTCT          ELSE, INCREMENT THE HYSTERESIS COUNTER,
530  0173  80               RTI                     AND WAIT FOR ANOTHER PULSE.
531                    **
532                    **
533                    ********************************************************************************
534                    **
535                    **                          CLOCKWISE ROUTINE.
536                    **
537       0174         CW      EQU     $
538                    **
539                    ***
540                    ***************  INCREMENT THE BINARY COUNTER (BINCT).  ******************
541                    ***
542  0174  B6 12              LDA     LOBIN   BEGIN AT THE LSB OF THE BINARY COUNTER.
543  0176  AB 01              ADD     #1      LOBIN = LOBIN + 1 ; CARRY -> C,CCR
544  0178  B7 12              STA     LOBIN
545                    ***
546  017A  B6 11              LDA     MIDBIN
547  017C  A9 00              ADC     #0              ADD THE CARRY TO THE MIDDLE BYTE.
548  017E  B7 11              STA     MIDBIN
549                    ***
550  0180  B6 10              LDA     HIBIN
551  0182  A9 00              ADC     #0              ADD THE CARRY TO THE HIGH BYTE.
552  0184  B7 10              STA     HIBIN
553                    ***
554                    ***
555                    ********************  CLR/SET MOD_32 APPROPRIATELY.  **********************
556                    ***
557                    ***     THE FOLLOWING SEVERAL LINES OF CODE ARE PRETTY MESSY.  ALL THAT
558                    ***     IS BEING DONE IS TO ENSURE THAT THE SCALE FACTOR IS SET PROPERLY.
559                    ***     FOR THE PAN AXIS THE SCALE FACTOR IS;
560                    ***
561                    ***                     1 PULSE => 0.007097 DEGREES
562                    ***
563  0186  B6 12              LDA     LOBIN           IF THE LOW FIVE BITS OF 'LOBIN' ARE NOT
564  0188  A4 1F              AND     #%00011111      ZERO THEN THE NUMBER ISN'T A MODULO 32 NUMBER.
565  018A  26 0F              BNE     NOT_32
566  018C  B6 11              LDA     MIDBIN          IF THE LOW SIX BITS OF 'MIDBIN' ARE ZERO
567  018E  27 07              BEQ     MOD             AND 'HIBIN' .NE. ZERO
568  0190  A4 3F              AND     #%00111111      THEN THE NUMBER IS MODULO 16,384, AND WE
569  0192  26 03              BNE     MOD             DON'T WANT TO SET 'MOD_32,STAT', UNLESS
570  0194  0D 11 04           BRCLR   B6,MIDBIN,NOT_32  THE NUMBER IS ALSO MODULO 32,768.
571                    ***
572  0197  1C 25       MOD     BSET    MOD_32,STAT
573  0199  20 02              BRA     DIRCHK
574                    ***
575  019B  1D 25       NOT_32  BCLR    MOD_32,STAT
576                    ***
577                    ***
578  019D  B6 10       DIRCHK  LDA     HIBIN
579  019F  2B 1D              BMI     CWNEG           IF HIBIN < 0 , WE'RE ROTATING CCW TOWARD
580                    ***                             THE ORIGIN.
581                    ***
582  01A1  26 14              BNE     CWPOS
583  01A3  B6 11              LDA     MIDBIN          ELSE IF BINCT .NE. 0
584  01A5  26 10              BNE     CWPOS           WE'RE ROTATING CW
585  01A7  B6 12              LDA     LOBIN           AWAY FROM THE ORIGIN.
586  01A9  26 0C              BNE     CWPOS
587
588                    ***                             ELSE, WE'VE ROTATED CW THROUGH THE ORIGIN.
589  01AB  15 25              BCLR    NEGTIV,STAT     CLR NEGATIVE SIGN.
```

```
590  01AD  AE 13              LDX     #PTR4   --
591  01AF  7F           CLRIT2 CLR     ,X        --
592  01B0  5C                  INCX                --RESET ALL COUNTERS AND DEGRES TO ZERO.
593  01B1  A3 1D                CPX     #THOUTH   --
594  01B3  23 FA                BLS     CLRIT2  --
595  01B5  20 6B                BRA     UPOUT         UPDATE OUTPUT.
596                       ***
597  01B7  AD 74          CWPOS BSR     ADDBCD
598  01B9  CD 02 6A             JSR     INCPOS
599  01BC  20 64                BRA     UPOUT
600                       ***
601  01BE  CD 02 CA       CWNEG JSR     SUBBCD
602  01C1  CD 02 6A             JSR     INCPOS
603  01C4  20 5C                BRA     UPOUT
604                       ***
605                       **
606                       ********************************************************************************
607                       **
608                       **                    COUNTER-CLOCKWISE ROUTINE.
609                       **
610       01C6            CCW     EQU     $
611                       **
612                       ***
613                       ******************  CLR/SET MOD_32 APPROPRIATELY.  **********************
614                       ***
615                       ***     AGAIN SET THE SCALE FACTOR TO;
616                       ***
617                       ***                     1 PULSE => 0.007097 DEGREES
618                       ***
619  01C6  B6 12                LDA     LOBIN         IF THE LOW FIVE BITS OF 'LOBIN' ARE NOT
620  01C8  A4 1F                AND     #%00011111    ZERO THEN THE NUMBER ISN'T A MODULO 32 NUMBER.
621  01CA  26 0F                BNE     NO_32
622  01CC  B6 11                LDA     MIDBIN        IF THE LOW SIX BITS OF 'MIDBIN' ARE ZERO
623  01CE  27 07                BEQ     MODLO         AND 'HIBIN' .NE. ZERO
624  01D0  A4 3F                AND     #%00111111    THEN THE NUMBER IS MODULO 16,384, AND WE
625  01D2  26 03                BNE     MODLO         DON'T WANT TO SET 'MOD_32,STAT', UNLESS
626  01D4  0D 11 04             BRCLR   B6,MIDBIN,NO_32  THE NUMBER IS ALSO MODULO 32,768.
627                       ***
628  01D7  1C 25          MODLO BSET    MOD_32,STAT
629  01D9  20 02                BRA     DECBIN
630                       ***
631  01DB  1D 25          NO_32 BCLR    MOD_32,STAT
632                       ***
633                       ***
634                       ***************  DECREMENT THE BINARY COUNTER (BINCT).  ****************
635                       ***
636  01DD  B6 12          DECBIN LDA    LOBIN   BEGIN AT THE LSB OF THE BINARY COUNTER.
637  01DF  A0 01                SUB     #1      LOBIN = LOBIN - 1 ; BORROW -> C,CCR
638  01E1  B7 12                STA     LOBIN
639                       ***
640  01E3  B6 11                LDA     MIDBIN
641  01E5  A2 00                SBC     #0      SUBTRACT THE CARRY FROM THE MIDDLE BYTE.
642  01E7  B7 11                STA     MIDBIN
643                       ***
644  01E9  B6 10                LDA     HIBIN
645  01EB  A2 00                SBC     #0      SUBTRACT THE CARRY FROM THE HIGH BYTE.
646  01ED  B7 10                STA     HIBIN
647                       ***
648  01EF  B6 10                LDA     HIBIN
649  01F1  2A 29                BPL     CCWPOS        IF HIBIN .GE. 0 , WE'RE ROTATING CCW TOWARD
650                       ***                          THE ORIGIN.
651  01F3  A6 FF                LDA     #-1     --
652  01F5  B1 10                CMP     HIBIN    --
653  01F7  26 1C                BNE     CCWNEG     --  ELSE IF BINCT .NE. -1 ,
654  01F9  B1 11                CMP     MIDBIN      -- WE'RE ROTATING CCW AWAY
655  01FB  26 18                BNE     CCWNEG     --  FROM THE ORIGIN.
```

```
01FD  B1 12          CMP    LOBIN   --
01FF  26 14          BNE    CCWNEG  --
                     ***
0201  14 25          BSET   NEGTIV,STAT   ELSE, WE'VE GONE THROUGH ORIGIN IN CCW
                     ***                  DIRECTION.  SET NEGATIVE SIGN.
                     ***                  AND SET ALL COUNTERS APPROPRIATELY.
0203  A6 03          LDA    #03     --
0205  B7 13          STA    PTR4      --
0207  B7 1A          STA    HUNDEG      -- DEGRES = 360.00
0209  A6 3C          LDA    #60       --
020B  B7 14          STA    PTR3    --
020D  AE 15          LDX    #PTR2   --
020F  7F             CLR    ,X        --
0210  5C             INCX               -- EVERYTHING ELSE IS ZERO BEFORE CHANGE.
0211  A3 19          CPX    #CTPTR1   --
0213  23 FA          BLS    CLREM   --
                     ***
0215  AD 16          CCWNEG BSR    ADDBCD
0217  CD 02 FD       JSR    DECPOS
021A  20 06          BRA    UPOUT
                     ***
021C  CD 02 CA       CCWPOS JSR    SUBBCD
021F  CD 02 FD       JSR    DECPOS
                     ***
                     ***
                     **
                     ********************************************************************************
                     **
                     ** OUTPUT ROUTINE. ROUTINE TO PRINT DATA TO THE OUTPUT PORTS.  BY
                     **       CALLING THE APPROPRIATE SUBROUTINE.('OUTCT' TO OUTPUT THE
                     **       THE COUNT AND 'OUTPOS' TO OUTPUT THE POSITION).
                     **
0222                 UPOUT  EQU    $
                     **
0222  09 25 04       BRCLR  POSCT,STAT,PUTCT
                     **
0225  CD 03 68       JSR    OUTPOS
0228  80             RTI
                     **
0229  CD 03 4E       PUTCT  JSR    OUTCT
022C  80             RTI
                     **
                     ********************************************************************************
                     ***********  SUBROUTINE TO INCREMENT THE BCD COUNTER (BCDCT).  *********
                     ***
022D  B6 19          ADDBCD LDA    CTPTR1
022F  AB 01          ADD    #1
0231  A1 63          CMP    #99     CTPTR > 99 ?
0233  23 05          BLS    OK1     NO, WE'RE OK HERE. LOOK UP THE FIRST TWO DIGITS.
0235  A0 64          SUB    #100    YES... MODIFY THE CTPTR,
0237  99             SEC            SET THE CARRY, AND
0238  20 01          BRA    OK1A    USE TABLE LOOK UP.
                     ***
023A  98             OK1    CLC                NO CARRY EXISTS IF WE ENTER AT THIS POINT.
023B  B7 19          OK1A   STA    CTPTR1  --
023D  97             TAX              --
                     ***                   -- LOOK UP THE TWO LEAST SIGNIFICANT DIGITS.
023E  D6 03 BE       LDA    TABLE,X   --
0241  B7 20          STA    TENONE  --
0243  24 24          BCC    NOMO           AND CONTINUE ONLY IF THERE WAS A CARRY.
                     ***
                     ***
0245  B6 18          LDA    CTPTR2
0247  A9 00          ADC    #0      ADD THE CARRY.
0249  A1 63          CMP    #99     CTPTR > 99 ?
024B  23 05          BLS    OK2     NO, WE'RE OK HERE. LOOK UP THE NEXT TWO DIGITS.
```

```
722  024D  A0 64              SUB   #100    YES... MODIFY THE CTPTR,
723  024F  99                 SEC           SET THE CARRY, AND
724  0250  20 01              BRA   OK2A    USE TABLE LOOK UP.
725                    ***
726  0252  98           OK2   CLC                   NO CARRY EXISTS IF WE ENTER AT THIS POINT.
727  0253  B7 18        OK2A  STA   CTPTR2  --
728  0255  97                 TAX             --
729                    ***                       -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
730  0256  D6 03 BE            LDA   TABLE,X  --
731  0259  B7 1F              STA   HUNDRD  --
732  025B  24 0C              BCC   NOMO            AND CONTINUE ONLY IF THERE WAS A CARRY.
733                    ***
734  025D  B6 17              LDA   CTPTR3
735  025F  A9 00              ADC   #0              ADD THE CARRY.
736  0261  B7 17              STA   CTPTR3  --
737  0263  97                 TAX             --
738                    ***                       -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
739  0264  D6 03 BE            LDA   TABLE,X  --
740  0267  B7 1E              STA   TENTHO  --
741                    ***
742  0269  81           NOMO  RTS
743                    ***
744                    ********************************************************************************
745                    *******  SUBROUTINE TO INCREMENT THE POSITION COUNTER (DEGRES).  *******
746                    ***
747                    ****  FIRST CHECK TO SEE IF THE BINARY COUNTER HAS REACHED A MODULO 32
748                    ****     NUMBER.
749                    ****
750  026A  0D 25 04    INCPOS BRCLR  MOD_32,STAT,INC7   'MOD_32,STAT' SET ?
751                    ****
752  026D  A6 08              LDA   #8                 YES ...
753  026F  20 02              BRA   INC             INCREMENT THE POSITION BY 0.008 DEGREES.
754                    ****
755  0271  A6 07        INC7  LDA   #7                 NO ....
756  0273  B7 22        INC   STA   POSINC          INCREMENT THE POSITION BY 0.007 DEGREES.
757                    ****
758                    **** ROUTINE TO INCREMENT THE POSITION COUNTER ,'DEGREES',BY A
759                    ****     PREDETERMINED AMOUNT, 'POSINC'.
760                    ****
761  0275  B6 16              LDA   PTR1
762  0277  BB 22              ADD   POSINC
763  0279  A1 09              CMP   #9      PTR1 > 9 ?
764  027B  23 05              BLS   OK3     NO, WE'RE OK HERE.  LOOK UP THE FIRST DIGIT.
765  027D  A0 0A              SUB   #10     YES... MODIFY THE CTPTR,
766  027F  99                 SEC           SET THE CARRY, AND
767  0280  20 01              BRA   OK3A    USE TABLE LOOK UP.
768                    ****
769  0282  98           OK3   CLC                   NO CARRY EXISTS IF WE ENTER AT THIS POINT.
770  0283  B7 16        OK3A  STA   PTR1    --
771  0285  97                 TAX               --
772                    ****                      -- LOOK UP THE LEAST SIGNIFICANT DIGIT.
773  0286  D6 03 BE            LDA   TABLE,X  --
774  0289  B7 1D              STA   THOUTH  --
775  028B  24 3C              BCC   DONE             AND CONTINUE ONLY IF THERE WAS A CARRY.
776                    ****
777                    ****
778  028D  B6 15              LDA   PTR2
779  028F  A9 00              ADC   #0      ADD THE CARRY.
780  0291  A1 63              CMP   #99     PTR2 > 99 ?
781  0293  23 05              BLS   OK4     NO, WE'RE OK HERE. LOOK UP THE NEXT TWO DIGITS.
782  0295  A0 64              SUB   #100    YES...MODIFY THE CTPTR,
783  0297  99                 SEC           SET THE CARRY,
784  0298  20 01              BRA   OK4A    AND USE TABLE LOOK UP.
785                    ****
786  029A  98           OK4   CLC                   NO CARRY EXISTS IF WE ENTER AT THIS POINT.
787  029B  B7 15        OK4A  STA   PTR2    --
```

```
0290  97          TAX             --
****                              -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
029E  D6 03 BE    LDA   TABLE,X   --
02A1  B7 1C       STA   HUNDTH    --
02A3  24 24       BCC   DONE            AND CONTINUE ONLY IF THERE WAS A CARRY.
****
****
02A5  B6 14       LDA   PTR3
02A7  A9 00       ADC   #0        ADD THE CARRY.
02A9  A1 63       CMP   #99       PTR3 > 99 ?
02AB  23 05       BLS   OK5       NO, WE'RE OK HERE. LOOK UP THE NEXT TWO DIGITS.
02AD  A0 64       SUB   #100      YES... MODIFY THE CTPTR,
02AF  99          SEC             SET THE CARRY, AND
02B0  20 01       BRA   OK5A      USE TABLE LOOK UP.
****
02B2  98          OK5   CLC                 NO CARRY EXISTS IF WE ENTER AT THIS POINT.
02B3  B7 14       OK5A  STA   PTR3    --
02B5  97                TAX           --
****                                -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
02B6  D6 03 BE          LDA   TABLE,X   --
02B9  B7 1B             STA   ONEDEG    --
02BB  24 0C             BCC   DONE            AND CONTINUE ONLY IF THERE WAS A CARRY.
****
02BD  B6 13             LDA   PTR4
02BF  A9 00             ADC   #0        ADD THE CARRY.
02C1  B7 13             STA   PTR4      --
02C3  97                TAX             --
****                                -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
02C4  D6 03 BE          LDA   TABLE,X   --
02C7  B7 1A             STA   HUNDEG    --
****
02C9  81          DONE  RTS
***
**
*****************************************************************************
******** SUBROUTINE TO DECREMENT THE BCD COUNTER (BCDCT). ************
***
02CA  B6 19       SUBBCD  LDA   CTPTR1
02CC  A0 01               SUB   #1        CTPTR > 99 ?
02CE  24 03               BCC   OK6       NO, WE'RE OK HERE. LOOK UP THE FIRST TWO DIGITS.
02D0  AB 64               ADD   #100      YES, MODIFY THE CTPTR, AND
02D2  99                  SEC             GENERATE A BORROW.
***
02D3  B7 19       OK6     STA   CTPTR1    --
02D5  97                  TAX             --
***                                 -- LOOK UP THE TWO LEAST SIGNIFICANT DIGITS.
02D6  D6 03 BE            LDA   TABLE,X   --
02D9  B7 20               STA   TENONE    --
02DB  24 1F               BCC   COMPLT        AND CONTINUE ONLY IF THERE WAS A BORROW.
***
***
02DD  B6 18               LDA   CTPTR2
02DF  A2 00               SBC   #0        SUBTRACT THE CARRY. CTPTR > 99 ?
02E1  24 03               BCC   OK7       NO, WE'RE OK HERE. LOOK UP THE NEXT TWO DIGITS.
02E3  AB 64               ADD   #100      YES, MODIFY THE CTPTR, AND
02E5  99                  SEC             GENERATE A BORROW.
***
02E6  B7 18       OK7     STA   CTPTR2    --
02E8  97                  TAX             --
***                                 -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
02E9  D6 03 BE            LDA   TABLE,X   --
02EC  B7 1F               STA   HUNDRD    --
02EE  24 0C               BCC   COMPLT          AND CONTINUE ONLY IF THERE WAS A CARRY.
***
02F0  B6 17               LDA   CTPTR3
02F2  A2 00               SBC   #0        SUBTRACT THE BORROW.
```

```
02F4  B7 17          STA   CTPTR3  --
02F6  97             TAX             --
***                                    -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
02F7  D6 03 BE       LDA   TABLE,X  --
02FA  B7 1E          STA   TENTHO  --
***
02FC  81             COMPLT RTS
***
*****************************************************************************
******* SUBROUTINE TO DECREMENT THE POSITION COUNTER (DEGRES). *******
***
****  FIRST CHECK TO SEE IF THE BINARY COUNTER HAS REACHED A MODULO 32
****    NUMBER.
****
02FD  00 25 04       DECPOS BRCLR  MOD_32,STAT,DEC7       'MOD_32,STAT' SET ?
****
0300  A6 08          LDA   #8                      YES...
0302  20 02          BRA   DEC           DECREMENT THE POSITION BY 0.008 DEGREES.
****
0304  A6 07          DEC7  LDA   #7                      NO ....
0306  B7 22          DEC   STA   POSINC        DECREMENT THE POSITION BY 0.007 DEGREES.
****
**** ROUTINE TO DECREMENT THE POSITION COUNTER ,'DEGREES',BY A
****    PREDETERMINED AMOUNT, 'POSINC'.
****
0308  B6 16          LDA   PTR1
030A  B0 22          SUB   POSINC  PTR1 < 0 ?
030C  24 03          BCC   OK8     NO, WE'RE OK HERE. LOOK UP THE FIRST DIGIT.
030E  AB 0A          ADD   #10     YES, MODIFY THE CTPTR, AND
0310  99             SEC           GENERATE A BORROW.
****
0311  B7 16          OK8   STA   PTR1    --
0313  97             TAX             --      .
****                                   -- LOOK UP THE LEAST SIGNIFICANT DIGIT.
0314  D6 03 BE       LDA   TABLE,X  --
0317  B7 1D          STA   THOUTH  --
0319  24 32          BCC   DUNSUB       AND CONTINUE ONLY IF THERE WAS A BORROW.
***
***
031B  B6 15          LDA   PTR2
031D  A2 00          SBC   #0      SUBTRACT THE BORROW.  PTR2 < 0 ?
031F  24 03          BCC   OK9     NO, WE'RE OK HERE. LOOK UP THE NEXT TWO DIGITS.
0321  AB 64          ADD   #100    YES, MODIFY THE CTPTR, AND
0323  99             SEC           GENERATE A BORROW.
***
0324  B7 15          OK9   STA   PTR2    --
0326  97             TAX             --
***                                    -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
0327  D6 03 BE       LDA   TABLE,X  --
032A  B7 1C          STA   HUNDTH  --
032C  24 1F          BCC   DUNSUB       AND CONTINUE ONLY IF THERE WAS A CARRY.
***
***
032E  B6 14          LDA   PTR3
0330  A2 00          SBC   #0      SUBTRACT THE BORROW.  PTR3 < 0 ?
0332  24 03          BCC   OK10    NO, WE'RE OK HERE. LOOK UP THE NEXT TWO DIGITS.
0334  AB 64          ADD   #100    YES, MODIFY THE CTPTR, AND
0336  99             SEC           GENERATE A BORROW.
****
0337  B7 14          OK10  STA   PTR3    --
0339  97             TAX             --
****                                   -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
033A  D6 03 BE       LDA   TABLE,X  --
033D  B7 1B          STA   ONEDEG  --
033F  24 0C          BCC   DUNSUB       AND CONTINUE ONLY IF THERE WAS A CARRY.
****
```

```
0341  B6 13              LDA   PTR4
0343  A2 00              SBC   #0      SUBTRACT THE BORROW.
0345  B7 13              STA   PTR4    --
0347  97                 TAX             --
                ****                        -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
0348  D6 03 BE           LDA   TABLE,X   --
034B  B7 1A              STA   HUNDEG  --
                ****
034D  81        DUNSUB   RTS
                ***
                **
                **
                ********************************************************************************
                **
                ** OUTPUT COUNT (OUTCT).  SUBROUTINE TO MOVE THE CURRENT COUNT (BCDCT)
                **       TO THE OUTPUT PORTS.  REMOVES THE DECIMAL POINT FROM THE
                **       DISPLAY AND BLANKS ALL BUT THE LEAST SIGNIFICANT DIGIT. ALSO
                **       SETS THE MINUS SIGN IF APPROPRIATE.
                **
034E    OUTCT   EQU   $
                **
034E  B6 20              LDA   TENONE
0350  B7 02              STA   PORTC
                **
0352  B6 1F              LDA   HUNDRD
0354  B7 00              STA   PORTA
                **
0356  B6 1E              LDA   TENTHO
0358  B7 01              STA   PORTB
                **
035A  1E 01              BSET  DECPT,PORTB
035C  19 01              BCLR  BLANK,PORTB
                **
035E  04 25 04           BRSET NEGTIV,STAT,MINUS
0361  1C 01              BSET  POSTIV,PORTB
0363  20 02              BRA   ALLDUN
                **
0365  1D 01     MINUS    BCLR  POSTIV,PORTB
                **
0367  81        ALLDUN   RTS
                **
                **
                ** OUTPUT POSITION (OUTPOS).  SUBROUTINE TO MOVE THE CURRENT POSITION
                **       COUNT (BCDCT) TO THE OUTPUT PORTS.  THE DECIMAL POINT IS
                **       DISPLAYED , AND ONLY THE MOST SIGNIFICANT DIGIT IS BLANKED.
                **
0368    OUTPOS   EQU   $
                **
0368  B6 1D              LDA   THOUTH  --
036A  A1 05              CMP   #5        --
036C  25 38              BLO   TRUNC       --    IF 5 > 'THOUTH' SIMPLY TRUNCATE THE
036E  B6 1C              LDA   HUNDTH          -- DISPLAY.  OTHERWISE...
0370  A4 09              AND   #9              -- IF THE LAST DIGIT ISN'T A NINE IT IS
0372  A1 09              CMP   #9              -- EASY TO ROUND UP.  JUST ADD A ONE.
0374  26 28              BNE   DECIMAL         --
0376  B6 1C              LDA   HUNDTH          -- BUT IF THE LAST DIGIT IS A NINE CHECK TO
0378  A1 99              CMP   #$99            -- SEE IF IT'S 99.  IF SO IT GETS GRIM.
037A  27 04              BEQ   UGLY            --
037C  AB 07              ADD   #7              -- IF THE NUMBER IS X9 AND X .NE. 9, THEN
037E  20 28              BRA   PCOUT           -- JUST ADD SEVEN TO ROUND UP. DUE TO
                **                            -- HEXIDECIMAL.
                **                            -- IF THE LOW TWO DIGITS ARE BOTH NINES
0380  A6 00     UGLY     LDA   #00           --   AND WE NEED TO ROUND UP....
0382  B7 02              STA   PORTC       --       MAKE THE LOW TWO DIGITS BOTH ZEROS
0384  99                 SEC             --       AND SET THE CARRY.
                **
```

```
 986  0385  B6 1B              LDA     ONEDEG  --      CHECK THE LAST DIGIT AS BEFORE.
 987  0387  A4 09              AND     #9        --    IF USING THIS PORTION OF THE CODE
 988  0389  A1 09              CMP     #9          --  THERE HAD TO BE A CARRY.
 989  038B  26 1E              BNE     NEXT          -- IF THE LAST DIGIT IS NOT A NINE USE THE
 990  038D  B6 1B              LDA     ONEDEG        -- ADC INSTRUCTION BELOW.
 991  038F  A1 99              CMP     #$99          -- IF IT IS A NINE, IS THE NEXT ONE A NINE
 992  0391  27 04              BEQ     RLUGLY        -- ALSO?
 993  0393  AB 07              ADD     #7          --  IF NOT JUST ADD SEVEN,
 994  0395  20 18              BRA     PAOUT     --    AND DISPLAY THE OUTPUT.
 995                       **
 996  0397  A6 00       RLUGLY LDA     #00       --
 997  0399  B7 00              STA     PORTA       --  IF SO ROUND UP THE MOST SIGNIFICANT
 998  039B  99                 SEC                 --  DIGIT AND SET EVERYTHING ELSE TO ZERO.
 999  039C  20 13              BRA     NEXT1     --
1000                       **
1001  039E  B6 1C       DECIMAL LDA    HUNDTH --
1002  03A0  AB 01              ADD     #1        --
1003                       **                        -- THIS IS ALL THAT NEEDS TO BE DONE IF
1004  03A2  B7 02              STA     PORTC       --  THE LAST DIGIT IS NOT A NINE.
1005  03A4  20 05              BRA     NEXT      --
1006                       **
1007  03A6  B6 1C       TRUNC  LDA     HUNDTH  --
1008  03A8  B7 02       PCOUT  STA     PORTC     --  AND IF THERE IS NO CARRY IT'S EVEN EASIER.
1009  03AA  98                 CLC               --
1010                       **
1011  03AB  B6 1B       NEXT   LDA     ONEDEG
1012  03AD  A9 00              ADC     #0
1013  03AF  B7 00       PAOUT  STA     PORTA
1014                       **
1015  03B1  B6 1A       NEXT1  LDA     HUNDEG
1016  03B3  A9 00              ADC     #0
1017  03B5  B7 01              STA     PORTB
1018                       **
1019  03B7  1C 01              BSET    POSTIV,PORTB
1020  03B9  18 01              BSET    BLANK,PORTB
1021  03BB  1F 01              BCLR    DECPT,PORTB
1022                       **
1023  03BD  81                 RTS
1024                       **
1025                       **
1026                       ********************************************************************************
1027                       **
1028                       ** SET UP THE TABLE TO BE USED WITH BCD INCREMENT/DECREMENT ROUTINES.
1029                       **
1030                       *       ENDS
1031                       *       DATA
1032  03BE  00 01 02 03 04 TABLE  FCB     $00,$01,$02,$03,$04,$05,$06,$07,$08,$09
      03C3  05 06 07 08 09
1033  03C8  10 11 12 13 14        FCB     $10,$11,$12,$13,$14,$15,$16,$17,$18,$19
      03CD  15 16 17 18 19
1034  03D2  20 21 22 23 24        FCB     $20,$21,$22,$23,$24,$25,$26,$27,$28,$29
      03D7  25 26 27 28 29
1035  03DC  30 31 32 33 34        FCB     $30,$31,$32,$33,$34,$35,$36,$37,$38,$39
      03E1  35 36 37 38 39
1036  03E6  40 41 42 43 44        FCB     $40,$41,$42,$43,$44,$45,$46,$47,$48,$49
      03EB  45 46 47 48 49
1037  03F0  50 51 52 53 54        FCB     $50,$51,$52,$53,$54,$55,$56,$57,$58,$59
      03F5  55 56 57 58 59
1038  03FA  60 61 62 63 64        FCB     $60,$61,$62,$63,$64,$65,$66,$67,$68,$69
      03FF  65 66 67 68 69
1039  0404  70 71 72 73 74        FCB     $70,$71,$72,$73,$74,$75,$76,$77,$78,$79
      0409  75 76 77 78 79
1040  040E  80 81 82 83 84        FCB     $80,$81,$82,$83,$84,$85,$86,$87,$88,$89
      0413  85 86 87 88 89
1041  0418  90 91 92 93 94        FCB     $90,$91,$92,$93,$94,$95,$96,$97,$98,$99
      041D  95 96 97 98 99
```

```
1042                        *       ENDS
1043                        *       CODE
1044                        ********************************************************************************
1045                        **
1046                        **                      SET UP MASK OPTION REGISTER.
1047                        **
1048  0422                          ABSOLUTE
1049                        **
1050                        **
1051  0F38                          ORG     MOR
1052  0F38  07                      FCB     $BIT2+BIT1+BIT0
1053                        **
1054                        **      COMMENTS:
1055                        **      BIT 7   CLOCK SOURCE  0 = CRYSTAL.
1056                        **      BIT 6   TIMER OPTION  0 = INTERNAL.
1057                        **      BIT 5   TIMER/CLOCK SOURCE  0 = INTERNAL.
1058                        **      BIT 4   NOT USED.
1059                        **      BIT 3   NOT USED.
1060                        **      BIT 2   SET  -
1061                        **      BIT 1   SET    -  PRESCALE SELECT  111 => 128
1062                        **      BIT 0   SET  -
1063                        **
1064                        **
1065                        ********************************************************************************
1066                        **
1067                        **                      ASSIGN INTERRUPT VECTORS.
1068                        **
1069  0FF8                          ORG     INTRPT
1070                        **
1071  0FF8  012E                    FDB     BLINK   TIMER/INT2 INTERRUPT VECTOR.
1072  0FFA  014D                    FDB     COUNT   EXTERNAL INTERRUPT VECTOR.
1073  0FFC  014D                    FDB     COUNT   SOFTWARE INTERRUPT VECTOR, NOT USED.
1074  0FFE  0080                    FDB     RESTRT  RESET VECTOR.
1075                        **
1076                        **
1077                        **
1078                          ENDS
1079  1000                    END


        Lines Assembled :  1079          Assembly Errors :  0
```

C. TILT

```
                        TTL                     POSITION DETERMINING PROGRAM (ELEVATION)
                *                       LATEST REVISION             9 MAY 89
                *                       FILE NAME                   TILT.ASM
                **
                ** PROGRAM DESCRIPTION
                **
                **
                **
                ** I/O REGISTER ADDRESSES
                **
0000            PORTA   EQU     $0000   I/O PORT A
0001            PORTB   EQU     $0001   I/O PORT B
0002            PORTC   EQU     $0002   I/O PORT C
0003            PORTD   EQU     $0003   INPUT PORT D
                **
                ** DATA DIRECTION REGISTER OFFSET
                **
0004            DDR     EQU     4       (eg. DDR FOR PORT A IS  PORTA+DDR )
                **
                ** OTHERS
                **
0008            TIMER   EQU     $0008   EIGHT BIT TIMER REGISTER.
0009            TCR     EQU     $0009   TIMER CONTROL REGISTER.
000A            MR      EQU     $000A   MISCELLANEOUS REGISTER.
0010            RAM     EQU     $0010   START OF ON-CHIP RAM(112 - 31 FOR STACK)
0080            ZROM    EQU     $0080   PAGE ZERO OF ROM.
0100            ROM     EQU     $0100   START OF MAIN ROM.
0F38            MOR     EQU     $0F38   MASK OPTION REGISTER.
0FF8            INTRPT  EQU     $0FF8   LOCATION OF INTERRUPT VECTORS.
1000            MEMSIZ  EQU     $1000   MEMORY ADDRESS SIZE.
                **
                ** EQUATES
                **
0001            BIT0    EQU     1
0002            BIT1    EQU     2
0004            BIT2    EQU     4
0008            BIT3    EQU     8
0010            BIT4    EQU     16
0020            BIT5    EQU     32
0040            BIT6    EQU     64
0080            BIT7    EQU     128
                **
0000            B0      EQU     0
0001            B1      EQU     1
0002            B2      EQU     2
0003            B3      EQU     3
0004            B4      EQU     4
0005            B5      EQU     5
0006            B6      EQU     6
0007            B7      EQU     7
                **
                ** EQUATES FOR THE TIMER CONTROL REGISTER
                **
                ***
0007            TIR     EQU     7       TIMER INTERRUPT REQUEST. 1 = REQUEST, 0 = NO REQ.
0006            TIM     EQU     6       TIMER INTERRUPT MASK. 1 = DISABLED, 0 = ENABLED.
0005            TIN     EQU     5       EXTERNAL OR INTERNAL CLOCK SOURCE. 1 = EXT, 0 = INT.
0004            TEE     EQU     4       EXTERNAL CLOCK ENABLE. NOT USED.
0003            PSC     EQU     3       PRESCALER CLEAR. NOT USED.
0002            PS2     EQU     2       (PS2)  --
0001            PS1     EQU     1       (PS1)    |--  PRESCALER SELECT BITS.
```

```
62          0000            PS0     EQU     0       (PS0) --
63                          **
64                          ** EQUATES FOR THE STATUS BYTE, 'STAT'.
65                          ***
66                          ***
67                          **
68          0007            UD      EQU     7       COUNT DIRECTION?  1 = UP, 0 = DOWN.
69          0006            MOD_32  EQU     6       IS 'BINCT' MODULO 32?   1 = YES, 0 = NO.
70          0005            FLASH   EQU     5       BLINK THE DISPLAY?  1 = YES, 0 = NO.
71          0004            POSCT   EQU     4       DISPLAY POSITION OR COUNT?  1 = POS, 0 = COUNT.
72          0003            L_SET   EQU     3       VALUE OF 'MODE,PORTD' LAST TIME.
73          0002            NEGTIV  EQU     2       IS 'BCDCT'NEGATIVE NUMBER?  1 = YES, 0 = NO.
74                          ***             1       NOT USED.
75                          ***             0       NOT USED.
76                          ***
77                          ***
78                          ** I/O EQUATES AND DESCRIPTIONS.
79                          ***
80                          ***
81                          ***      PORT A  (I/O)
82                          ***
83                          ***     +-------+-------+-------+-------+-------+-------+-------+-------+
84                          ***     |          BCD DIGIT #4         |          BCD DIGIT #3         |
85                          ***     +-------+-------+-------+-------+-------+-------+-------+-------+
86                          ***     |  D4   |  C4   |  B4   |  A4   |  D3   |  C3   |  B3   |  A3   |
87                          ***     +-------+-------+-------+-------+-------+-------+-------+-------+
88                          *** BIT     7       6       5       4       3       2       1       0
89                          ***
90                          ***
91                          ***      PORT B  (I/O)
92                          ***
93                          ***     +-------+-------+-------+-------+-------+-------+-------+-------+
94                          ***     |        DISPLAY CONTROL        |           NOT USED            |
95                          ***     +-------+-------+-------+-------+-------+-------+-------+-------+
96                          ***     | DECPT | POSTIV|       | BLANK |       |       |       |       |
97                          ***     +-------+-------+-------+-------+-------+-------+-------+-------+
98                          *** BIT     7       6       5       4       3       2       1       0
99                          ***
100         0007            DECPT   EQU     7       TO DISPLAY THE DECIMAL POINT...DECPT IS CLEARED
101         0006            POSTIV  EQU     6       USED TO DISPLAY NEGATIVE SIGN...CLEARED TO SHOW
102                         ***                     MINUS SIGN.
103         0004            BLANK   EQU     4       TO BLANK DIGITS 2 AND 3...CLEAR BLANK.
104                         ***                     DIGITS 4 AND 5 ARE ALWAYS BLANKED.
105                         ***                     DIGIT 1 IS NEVER BLANKED.
106                         ***
107                         ***      PORT C  (I/O)
108                         ***
109                         ***     +-------+-------+-------+-------+-------+-------+-------+-------+
110                         ***     |          BCD DIGIT #2         |BCD DIGIT #1(LEAST SIGNIFICANT)|
111                         ***     +-------+-------+-------+-------+-------+-------+-------+-------+
112                         ***     |  D2   |  C2   |  B2   |  A2   |  D1   |  C1   |  B1   |  A1   |
113                         ***     +-------+-------+-------+-------+-------+-------+-------+-------+
114                         *** BIT     7       6       5       4       3       2       1       0
115                         ***
116                         ***
117                         ***      PORT D  (INPUT ONLY)
118                         ***
119                         ***     +-------+-------+-------+-------+-------+-------+-------+-------+
120                         ***     | CH_A  | INT2  | CH_B  | FUNCT |  SET  |       |       |       |
121                         ***     +-------+-------+-------+-------+-------+-------+-------+-------+
122                         *** BIT     7       6       5       4       3       2       1       0
123                         ***
124         0007            CH_A    EQU     7       INDICATES THE STATUS OF CHANNEL A.
125         0006            INT2    EQU     6       INTERRUPT #2.  USED TO CHANGE DISPLAY MODES.
126         0005            CH_B    EQU     5       INDICATES THE STATUS OF CHANNEL B.
127         0004            FUNCT   EQU     4       USED TO PUT THE PROGRAM IN A MODE THAT WILL ALLOW
```

```
                        ***                     'HYST' TO BE INCREMENTED.
0003            SET     EQU     3       INCREMENTS 'HYST' WHEN TOGGLED AND FUNCT IS LOW.
                        ***
                        ***
                        ********************************************************************************
                        **                                                                              *
                        **                          RAM VARIABLES                                       *
                        **                                                                              *
                        ********************************************************************************
                        **
                        ** RESERVE MEMORY SPACE FOR THE PROGRAM VARIABLES.
                        **
0000                            DATA
                        **
                        **
0000                            ABSOLUTE (ABSOLUTE ADDRESSING USED HERE TO RELATIVE DIRECTIVE)
                        **
0010                            ORG     RAM     START OF RAM.
                        **
                        *** BINARY COUNT.
0010            BINCT   RMB     2
0010            HIBIN   EQU     BINCT   HI BYTE.
0011            LOBIN   EQU     BINCT+1 LO BYTE.
                        **
                        *** POSITION POINTERS.
0012            PTR     RMB     3       EACH BYTE POINTS TO A POSITION IN THE
                        **              TABLE THAT CONTAINS ONE OR TWO DIGITS
                        **              OF THE BCD POSITION.
0012            PTR3    EQU       PTR   MOST SIGNIFICANT DIGITS.
0013            PTR2    EQU       PTR+1
0014            PTR1    EQU       PTR+2 LEAST SIGNIFICANT DIGIT.
                        **
                        *** COUNT POINTERS.
0015            CTPTR   RMB     2       EACH BYTE POINTS TO A POSITION IN THE
                        **              TABLE THAT CONTAINS TWO OF THE DIGITS
                        **              IN THE BCD COUNT.
0015            CTPTR2  EQU     CTPTR   MOST SIGNIFICANT DIGITS.
0016            CTPTR1  EQU     CTPTR+1 LEAST SIGNIFICANT DIGITS.
                        **
                        *** BCD POSITION IN DEGREES.
0017            DEGRES  RMB     3
0017            ONEDEG  EQU     DEGRES        CONTENTS X   1.000
0018            HUNDTH  EQU     DEGRES+1      CONTENTS X   0.010
0019            THOUTH  EQU     DEGRES+2    + CONTENTS X   0.001
                        ***                   --------------------
                        ***                     POSITION IN DEGREES
                        **
                        *** BCD COUNT.
001A            BCDCT   RMB     2
001A            HUNDRD  EQU     BCDCT         CONTENTS X    100
001B            TENONE  EQU     BCDCT+1     + CONTENTS X      1
                        ***                   ----------------------
                        ***                   NUMBER OF PULSES COUNTED
                        ***
                        *** HYSTERESIS COUNTER. POINTS TO A NUMBER IN THE TABLE THAT IS THE
                        ***      AMOUNT OF HYSTERESIS PRESENT IN THE SYSTEM.  INITIALIZED TO 6.
001C            HYSTPT  RMB      1
                        **
                        *** POSITION INCREMENT.  CONTAINS A NUMBER, THAT WHEN MULTIPLIED BY 0.001
                        ***      IS THE NUMBER OF DEGREES THAT THE POSITION COUNTER (BCDPOS) IS
                        ***      TO BE INCREMENTED OR DECREMENTED DURING PROGRAM EXECUTION.
                        ***      THE VALUE OF 'POSINC', DETERMINED EXPERIMENTALLY, SHOULD BE
                        ***      7.0452.  SINCE THE PROGRAM IS DESIGNED WORK WITH INTEGERS ONLY
                        ***      THIS NUMBER IS ROUNDED TO 7.  TO REDUCE THE CUMULATIVE EFFECT OF
                        ***      THE ROUND OFF, EVERY 32 COUNTS 'POSINC' IS SET EQUAL TO 8.  THIS
                        ***      AGAIN LEADS TO SOME CUMULATIVE ERROR, BUT THE SMALL ANGULAR RANGE
```

```
                        ***     OF THE TILT ANGLE (10.5 DEGREES) ALLOWS US TO NEGLECT ANY FURTHER
                        ***     MODIFICATIONS.
                        ***
001D                    POSINC  RMB     1
                        ***
                        *** HYSTERESIS VARIABLES.  USED TO ELIMINATE THE EFFECTS OF BACKLASH ON
                        ***     THE POSITION MEASUREMENTS.
                        ***
001E                    HYST    RMB     1       THE THRESHHOLD VALUE DETERMINED
                        ***                     EXPERIMENTALLY.
001F                    HYSTCT  RMB     1       CURRENT AMOUNT OF HYSTERESIS MEASURED.
                        ***
                        *** STATUS BYTE.  USED TO KEEP TRACK OF WHAT IS GOING ON.
                        ***
0020                    STAT    RMB     1       CURRENT STATUS.
0021                    LSTAT   RMB     1       PREVIOUS/LAST STATUS.  USED TO KEEP TRACK OF
                        ***                     L_SET ONLY.
                        ***
                        *** TIMER COUNTER.  USED IN CONJUNCTION WITH THE TIMER PRESCALER AND THE
                        ***     TDR TO KEEP TRACK OF ONE SEC. INTERVALS.  USED IN BLINKING THE
                        ***     DISPLAY.  INITIALLY SET TO 31, WHEN THE 'FLASH' BIT OF 'STAT'
                        ***     IS SET.  TIMCT IS DECREMENTED EACH CLOCK INTERRUPT (APPROX. 31
                        ***     TIMES PER SEC).  RESET TO 31 WHEN CONTENTS GO TO ZERO.
                        ***     WHEN (TIMCT)=0 THE DISPLAY WILL TOGGLE.
                        ***
0022                    TIMCT   RMB     1
                        **
                                ENDS
                        **
                        ***
                        ********************************************************************************
                        **                                                                             *
                        **                          PAGE ZERO ROM
                        **                                                                             *
                        ********************************************************************************
                        **
                        **                  INITIALIZATION ROUTINE.
                        **
                        **
                                CODE
                        **
0080                            ORG     ZROM    PAGE ZERO ROM.
                        **
0080                            RELATIVE        RELATIVE ADDRESSING MUST BE USED FOR THE BRANCH.
                        **
        0080            RESTRT  EQU     $       THIS IS THE ENTRY POINT WHEN AN EXTERNAL
                        ***                     INTTERRUPT OCCURS.
                        **
                        ********************************************************************************
                        **
                        ***             INITIALIZE THE PC AND CLEAR RAM.
                        ***
                        ***
0080    9B                      SEI             SET INTERUPT TO AVOID INTERUPTION AND
0081    9C                      RSP             RESET THE STACK POINTER. JUST IN CASE!
                        ***
0082    AE 10                   LDX     #BINCT  CLEAR ALL OF THE VARIABLES BETWEEN
0084    7F              CLRIT   CLR     ,X      'BINCT' AND 'TIMCT' (INCLUSIVE).NOTE
0085    5C                      INCX            THAT THIS SETS THE COUNTER AND THE POS-
0086    A3 22                   CPX     #TIMCT  ITION TO ZERO. THIS MEANS THAT ROTATION
0088    23 FA                   BLS     CLRIT   SHOULD START IN AN INCREASING (CW)
                        ***                     DIRECTION FROM THE MOST CCW POSITION
                        ***                     AFTER A RESET.
                        ***
008A                            ABSOLUTE        BACK TO ABSOLUTE ADDRESSING.
                        ***
```

```
                        ********************************************************************************
                        ***
                        ***                          ESTABLISH I/O PORTS.
                        ***
008A  A6 FF                     LDA     #-1         PORTS A,B,C ARE CONFIGURED AS
008C  B7 04                     STA     PORTA+DDR   ALL OUTPUT. PORT D IS ALL INPUT
008E  B7 05                     STA     PORTB+DDR   SO THERE IS NO MASK TO SET.
0090  B7 06                     STA     PORTC+DDR
                        ***
                        ********************************************************************************
                        ***                  SET UP THE STATUS REGISTER.
                        ***
0092  A6 08                     LDA     #%00001000  --
0094  B4 03                     AND     PORTD         |--> SET UP 'L_SET' BIT OF 'STAT'.
0096  B7 20                     STA     STAT        --
                        ***
0098  1C 20                     BSET    MOD_32,STAT  0 IS MODULO 32.
                        **
                        ********************************************************************************
                        ***
                        ***                      INITIALIZE HYSTCT.
                        ***
009A  A6 06                     LDA     #06
009C  B7 1E                     STA     HYST
009E  B7 1C                     STA     HYSTPT
                        ***
                        ********************************************************************************
                        **
                        **        SET UP THE TIMER FOR A 4 MHZ CRYSTAL / 4 = 1 MHZ CLOCK.
                        ***
                        ***     NOTE:  THE MASK OPTION REGISTER IS IN ROM.  IT IS SET UP AT
                        ***            THE END OF THE PROGRAM.
                        ***
                        **********************   SET UP THE TCR.  **********************************
                        ***
00A0  A6 47                     LDA       #BIT6+BIT2+BIT1+BIT0
                        ***              (TIM)|(PS2)(PS1)(PS0)
                        ***(DISABLE INTERRUPT)|(PRESCALE BY 128)
                        ***
00A2  B7 09                     STA     TCR
                        ***
                        ***
                        *** SET UP THE TIMER.
                        ***
00A4  A6 FF                     LDA     #255    1 MHZ/(128*255) = 30.6 (APPROX. 31)
00A6  B7 08                     STA TIMER
                        ***
                        ****************  INITIALIZE THE TIMER COUNTER.  ************************
                        ***
00A8  A6 1F                     LDA     #31     PROVIDES FOR 1 SEC. BLINK INTERVAL.
00AA  B7 22                     STA     TIMCT   FOR 2 SEC. INTERVAL JUST USE TIMECT=62, etc.
                        ***
                        ***
                        ********************************************************************************
                        **
                        **                      SET UP MISCELLANEOUS REGISTER.
                        **
00AC  1D 0A                    BCLR    B6,MR    ENABLES THE SECOND INTERRUPT.
                        **
                        ********************************************************************************
                        **
                        **                   COUNT = 0 IS DISPLAYED INITIALLY.
                        **
00AE  CD 02 D8                  JSR     OUTCT
00B1  1C 01                     BSET    POSTIV,PORTB
                        **
```

```
326                             **
327  0083  9A                   *       CLI             CLEAR THE INTERRUPT MASK TO GET STARTED.
328                             ***
329  0084                       *       RELATIVE        RELATIVE ADDRESSING MUST BE USED FOR THE
330                             ***                     REMAINDER OF THE PROGRAM.
331                             ***
332                             ********************************************************************************
333                             **
334                             ** WAIT LOOP.  EXECUTES, UNTIL AN INTERRUPT OCCURS.
335                             **
336  0084  09 03 08             PAUSE   BRCLR   FUNCT,PORTD,CHHYST WANT TO CHANGE HYST?
337                             **                                 YES...GO TO CHHYST.
338  0087  06 03 04                     BRSET   SET,PORTD,SBIT     NO...'SET,PORTD' SET?
339  008A  17 20                        BCLR    L_SET,STAT              NO...CLEAR 'L_SET,STAT'
340  008C  20 F6                        BRA     PAUSE                     AND LOOP.
341  008E  16 20                SBIT    BSET    L_SET,STAT              YES...SET 'L_SET,STAT'
342  0090  20 F2                        BRA     PAUSE                     AND LOOP.
343                             **
344                             ********************************************************************************
345                             **
346                             ** HYSTERESIS MODIFICATION ROUTINE.  PERMITS MODIFICATION OF THE
347                             ** HYSTERESIS BUFFER WITHOUT REPROGRAMMING.
348                             **                                 YES...
349  0092  A6 40                CHHYST  LDA     #BIT6              DISABLE TIMER INTERRUPT.
350  0094  B7 09                        STA     TCR
351                             ***
352  0096  A6 08                        LDA     #%00001000         SAVE 'L_SET'
353  0098  B4 20                        AND     STAT               INTO
354  009A  B7 21                        STA     LSTAT              'LSTAT'.
355  009C  A6 08                        LDA     #%00001000
356  009E  B4 03                        AND     PORTD              'SET,PORTD' --> ACCUMULATOR
357  00A0  B1 21                        CMP     LSTAT              HAS THE SET SWITCH BEEN CHANGED?
358  00A2  27 15                        BEQ     DISPLA
359                             ***                                YES...
360  00A4  A6 08                        LDA     #%00001000         --
361  00A6  B8 20                        EOR     STAT                 |-->CHANGE 'L_SET,STAT',
362  00A8  B7 20                        STA     STAT               --
363  00AA  3C 1E                        INC     HYST               THEN INCREMENT THE HYSTERESIS
364  00AC  3C 1C                        INC     HYSTPT             POINTER AND 'HYST'...
365                             ***
366  00AE  B6 1C                        LDA     HYSTPT           --
367  00B0  A1 19                        CMP     #25                --
368  00B2  23 05                        BLS     DISPLA               --
369                             ***                                   -- BUT NOT ABOVE 25.
370  00B4  4F                           CLRA                         --       THEN--->|
371  00B5  B7 1E                        STA     HYST               --                |
372  00B7  B7 1C                        STA     HYSTPT           --                  |
373                             ***                                                  |
374  00B9  A6 00                DISPLA  LDA     #%00000000         NO... JUST--------->|
375  00BB  B7 00                        STA     PORTA            --                  |
376  00BD  A6 C0                        LDA     #%11000000         --                |
377  00BF  B7 01                        STA     PORTB                --             \|/
378                             ***                                      -- DISPLAY CURRENT 'HYST'.
379  00C1  BE 1C                        LDX     HYSTPT               --
380  00C3  D6 03 28                     LDA     TABLE,X            --
381  00C6  B7 02                        STA     PORTC            --
382                             ***                                IS 'HYST' SETTING COMPLETE?
383  00C8  09 03 C7                     BRCLR   FUNCT,PORTD,CHHYST NO... KEEP CHECKING 'SET'.
384  00CB  04 20 04                     BRSET   NEGTIV,STAT,SIGN   YES... RESET THE DISPLAY.
385  00CE  1C 01                        BSET    POSTIV,PORTB
386  00D0  20 02                        BRA     DIR
387  00D2  1D 01                SIGN    BCLR    POSTIV,PORTB
388  00D4  08 20 05             DIR     BRSET   POSCT,STAT,SHOPOS
389                             **
390  00D7  CD 02 D8                     JSR     OUTCT
391  00DA  20 03                        BRA     DUNCHG
```

```
392                             ***
393  010C  CD 02 E5     SHOPOS  JSR     OUTPOS
394                             ***
395  010F  0B 20 04     DUNCHG  BRCLR   FLASH,STAT,NO_INT  IF THE DISPLAY IS TO BLINK...
396  0112  A6 07                LDA     #BIT2+BIT1+BIT0   ENABLE TIMER INTERRUPT AND RESET
397  0114  B7 09                STA     TCR               TIMER PRESCALER
398  0116  20 9C          NO_INT  BRA     PAUSE             PRIOR TO RETURNING.
399                             ***
400                             **
401                             ********************************************************************************
402                             ** MAXIMUM EXECUTION TIME FOR THE REMAINDER OF THE PROGRAM OCCURS
403                             ** IF THE COUNTER ROTATES THROUGH ZERO AS THE DISPLAY MODE IS CHANGED
404                             ** FROM THE BLINKING MODE TO THE COUNT MODE AT THE SAME TIME THAT THE
405                             ** BLINKING ROUTINE IS CAUSING THE DISPLAY TO TOGGLE TO SHOW THE
406                             ** POSITION IN DEGREES.
407                             **     MAXIMUM EXECUTION TIME = 67 + 140 + 618 = 825 CLOCK CYCLES.
408                             **
409                             ********************************************************************************
410                             **
411                             **MODE CHANGE ROUTINE.  CHANGES THE DISPLAY MODE FROM
412                             **       COUNT -> POSITION -> BLINKING -> COUNT ->....(ETC.)
413                             **       MAXIMUM EXECUTION TIME OF 127 CLOCK CYCLES OCCURS WHEN THE
414                             **       DISPLAY MODE IS CHANGED FROM DISPLAYING THE COUNT TO DISPLAYING
415                             **       THE POSITION (IN DEGREES).
416                             **       IF THE DISPLAY IS CHANGED FROM BLINKING TO A COUNT DISPLAY
417                             **       EXECUTION TIME IS 67 CLOCK CYCLES.
418                             **
419  0118  1F 0A          CHMODE  BCLR    B7,MR             AVOID REPEATED INTERRUPTS.
420                             **
421  011A  0A 20 10             BRSET   FLASH,STAT,DIS_CT IF FLASHING, DISPLAY COUNT...
422  011D  09 20 07             BRCLR   POSCT,STAT,DISPOS IF SHOWING COUNT, DISPLAY POSITION...
423  0120  1A 20                BSET    FLASH,STAT        ELSE, BLINK.
424                             **
425  0122  A6 07                LDA     #BIT2+BIT1+BIT0   ENABLE TIMER INTERRUPT AND RESET
426  0124  B7 09                STA     TCR               TIMER PRESCALER.
427  0126  80                  RTI
428                             **
429  0127  18 20          DISPOS  BSET    POSCT,STAT      --
430  0129  CD 02 E5             JSR     OUTPOS            |-- DISPLAY CURRENT POSITION, AND WAIT.
431  012C  80                   RTI                     --
432                             **
433  012D  A6 47          DIS_CT  LDA     #BIT6+BIT2+BIT1+BIT0  DISABLE TIMER INTERRUPT AND RESET
434  012F  B7 09                STA     TCR                   TIMER PRESCALER.
435  0131  19 20                BCLR    POSCT,STAT      --
436  0133  1B 20                BCLR    FLASH,STAT        |
437                             **                        |-- DISPLAY CURRENT COUNT, AND WAIT.
438  0135  CD 02 D8             JSR     OUTCT             |
439  0138  80                   RTI                     --
440                             **
441                             ********************************************************************************
442                             **
443                             ** BLINK ROUTINE.  INTERRUPT ROUTINE TO CHANGE THE DISPLAY FROM POSITION
444                             **       TO COUNT OR VICE VERSA EVERY 31 ST TIMER INTERRUPT IF THE
445                             **       'FLASH' BIT OF 'STAT' IS SET.
446                             **       MAXIMUM EXECUTION TIME OF 140 CLOCK CYCLES OCCURS WHEN THE
447                             **       DISPLAY IS TOGGLED FROM A COUNT DISPLAY TO A POSITION DISPLAY.
448                             **
449       0139          BLINK   EQU     $
450  0139  0F 09 DC             BRCLR   TIR,TCR,CHMODE        IF THE INTERRUPT WAS NOT A TIMER
451                             **                            INTERRUPT IT MUST BE FROM INT2.
452                             **
453  013C  1F 09                BCLR    TIR,TCR               AVOID REPEATED TIMER INTERRUPTS.
454                             **
455  013E  3A 22                DEC     TIMCT                 IF THERE HAVE BEEN 31 TIMER
456  0140  27 01                BEQ     CHGDIS                INTERRUPTS  (1 SEC), IT'S TIME TO
457                             **                            CHANGE THE DISPLAY.
```

```
0142  80                    RTI                          OTHERWISE, IT'S BACK TO WORK.
                      **
0143  A6 1F           CHGDIS LDA    #31                   RESET TIMCT TO 31 (1 SEC. INTERVAL).
0145  B7 22                 STA    TIMCT
                      **
0147  B6 20                 LDA    STAT          --
0149  A8 10                 EOR    #%00010000      |-- CHANGE 'POSCT' BIT OF 'STAT'.
014B  B7 20                 STA    STAT          --
                      **
014D  08 20 04              BRSET  POSCT,STAT,POSOUT    DECIDE ON CORRECT DISPLAY.
                      **
                      ************ CHANGE THE DISPLAY TO SHOW THE COUNT..... ***************
                      **
0150  CD 02 D8              JSR    OUTCT
0153  80                    RTI
                      **
                      ************ OR HAVE THE DISPLAY SHOW THE POSITION. ******************
                      **
0154  CD 02 E5       POSOUT JSR    OUTPOS
0157  80                    RTI
                      **
                      ********************************************************************************
                      **
                      **                          COUNT ROUTINE.
                      **        WHEN A COUNT IS RECEIVED THIS IS THE ENTRY POINT .
                      **        MAXIMUM EXECUTION TIME OF 618 CLOCK CYCLES OCCURS WHEN THE
                      **        COUNTER ROTATES CCW THROUGH ZERO AND THE POSITION (IN DEGREES)
                      **        IS BEING DISPLAYED.
                      **
                      **
                      ** CURRENT DIRECTION OF ROTATION IS DETERMINED BY INSPECTING THE STATUS
                      ** OF 'CH_A' AND 'CH_B'.  THE FOUR POSSIBILITIES AND THE ASSOCIATED
                      ** DIRECTION OF ROTATION ARE AS SHOWN BELOW.  NOTE THAT THIS SCHEME
                      ** PREVENTS MULTIPLE OSCILLATIONS ABOUT A SINGLE POINT FROM
                      ** REPEATEDLY INCREMENTING OR DECREMENTING THE COUNTER.
                      **
                      **   +--------+--------+-----------------+------------------+
                      **   |  CH_A  |  CH_B  |   DIRECTION     |  COUNT THE PULSE? |
                      **   |        |        |   OF ROTATION   |                   |
                      **   +--------+--------+-----------------+------------------+
                      **   |   LO   |   LO   |       CW        |        NO         |
                      **   |   LO   |   HI   |      CCW        |        YES        |
                      **   |   HI   |   LO   |      CCW        |        NO         |
                      **   |   HI   |   HI   |       CW        |        YES        |
                      **   +--------+--------+-----------------+------------------+
                      **
                      ******** FIRST SEE IF WE ARE SUPPOSED TO COUNT THIS INTERUPT. ********
                      **
0158           COUNT  EQU    $
0158  0A 03 01              BRSET  CH_B,PORTD,OKCT     IF CH_B IS LO WE DON'T COUNT THE
015B  80                    RTI                        INTERRUPT.
                      **
                      ************** IF THE INTERRUPT IS VALID UPDATE 'STAT'. ***************
                      **
015C  A6 7F           OKCT  LDA    #%01111111
015E  B4 20                 AND    STAT               SAVE ALL OF THE OLD 'STAT' EXCEPT THE
0160  B7 20                 STA    STAT               DIRECTION OF ROTATION.
0162  A6 80                 LDA    #%10000000
0164  B4 03                 AND    PORTD              'CH_A,PORTD' INDICATES THE DIRECTION
                      *                               OF ROTATION AND BECOMES 'UD,STAT'.
0166  BA 20                 ORA    STAT               ADD THE RESULTS TO GET
0168  B7 20                 STA    STAT               THE NEW 'STAT'.
                      **
                      ** DECIDE IF THE "SLACK" DUE TO BACKLASH/HYSTERESIS HAS BEEN TAKEN OUT.
                      **
016A  0E 20 09              BRSET  UD,STAT,HYSTCK  IF ROTATING CW SEE BELOW.
```

```
524  016D  B6 1F              LDA     HYSTCT          ELSE, SEE IF WE DECREMENT THIS TIME.
525  016F  27 4B              BEQ     CCW             IF HYSTCT=0, GO TO THE COUNT DOWN
526                    **                             ROUTINE.
527  0171  A0 01              SUB     #1              ELSE, DECREMENT THE HYSTERESIS COUNTER,
528  0173  B7 1F              STA     HYSTCT
529  0175  80                RTI                     AND WAIT FOR THE NEXT INTERRUPT.
530                    **
531  0176  B6 1E    HYSTCK    LDA     HYST            IF ROTATING CW....
532  0178  B1 1F              CMP     HYSTCT          AND  HYST = HYSTCT ....
533  017A  27 03              BEQ     CW              COUNT THE PULSE .
534  017C  3C 1F              INC     HYSTCT          ELSE, INCREMENT THE HYSTERESIS COUNTER,
535  017E  80                RTI                     AND WAIT FOR ANOTHER PULSE.
536                    **
537                    **
538                    ********************************************************************************
539                    **
540                    **                          CLOCKWISE ROUTINE.
541                    **
542       017F         CW      EQU     $
543                    **
544                    ***
545                    ***************  INCREMENT THE BINARY COUNTER (BINCT).  ******************
546                    ***
547  017F  B6 11              LDA     LOBIN   BEGIN AT THE LSB OF THE BINARY COUNTER.
548  0181  AB 01              ADD     #1      LOBIN = LOBIN + 1 ; CARRY -> C,CCR
549  0183  B7 11              STA     LOBIN
550                    ***
551  0185  B6 10              LDA     HIBIN
552  0187  A9 00              ADC     #0              ADD THE CARRY TO THE HIGH BYTE.
553  0189  B7 10              STA     HIBIN
554                    ***
555                    ***
556                    ******************  CLR/SET MOD_32 APPROPRIATELY.  **********************
557                    ***
558  018B  B6 11              LDA     LOBIN         IF THE LOW FIVE BITS OF 'LOBIN' ARE NOT
559  018D  A4 1F              AND     #%00011111    NOT THEN THE NUMBER ISN'T A MODULO 32 NUMBER.
560  018F  26 04              BNE     NOT_32
561                    ***
562  0191  1C 20              BSET    MOD_32,STAT
563  0193  20 02              BRA     DIRCHK
564                    ***
565  0195  1D 20    NOT_32    BCLR    MOD_32,STAT
566                    ***
567                    ***
568  0197  B6 10    DIRCHK    LDA     HIBIN
569  0199  2B 19              BMI     CWNEG           IF HIBIN < 0 , WE'RE ROTATING CW TOWARD
570                    ***                            THE ORIGIN.
571                    ***
572  019B  26 10              BNE     CWPOS           ELSE IF BINCT .NE. 0 ,
573  019D  B6 11              LDA     LOBIN           WE'RE ROTATING CW
574  019F  26 0C              BNE     CWPOS           AWAY FROM THE ORIGIN.
575                    ***
576  01A1  15 20              BCLR    NEGTIV,STAT     ELSE, WE'VE GONE THROUGH ORIGIN IN CW
577                    ***                            DIRECTION.  CLR NEGATIVE SIGN.
578  01A3  AE 12              LDX     #PTR3   --
579  01A5  7F        CLRIT2    CLR     ,X        --
580  01A6  5C                 INCX                --RESET ALL COUNTERS AND DEGRES TO ZERO.
581  01A7  A3 19              CPX     #THOUTH   --
582  01A9  23 FA              BLS     CLRIT2  --
583  01AB  20 4C              BRA     UPOUT           UPDATE OUTPUT.
584                    ***
585  01AD  AD 5E    CWPOS     BSR     ADDBCD
586  01AF  CD 02 32           JSR     INCPOS
587  01B2  20 45              BRA     UPOUT
588                    ***
589  01B4  CD 02 7A    CWNEG   JSR     SUBBCD
```

```
590  01B7  CD 02 9A          JSR     DECPOS
591  01BA  20 3D             BRA     UPOUT
592                  ***
593                  **
594                  ********************************************************************************
595                  **
596                  **                  COUNTER-CLOCKWISE ROUTINE.
597                  **
598       018C       CCW     EQU     $
599                  **
600                  ***
601                  ******************  CLR/SET MOD_32 APPROPRIATELY.  **********************
602                  ***
603  01BC  B6 11             LDA     LOBIN       ;IF THE LOW FIVE BITS OF 'LOBIN' ARE NOT
604  01BE  A4 1F             AND     #%00011111  ZERO THEN THE NUMBER ISN'T A MODULO 32 NUMBER.
605  01C0  26 04             BNE     NO_32
606                  ***
607  01C2  1C 20             BSET    MOD_32,STAT
608  01C4  20 02             BRA     DECBIN
609                  ***
610  01C6  1D 20     NO_32   BCLR    MOD_32,STAT
611                  ***
612                  ***
613                  ****************  DECREMENT THE BINARY COUNTER (BINCT).  ****************
614                  ***
615  01C8  B6 11     DECBIN  LDA     LOBIN   BEGIN AT THE LSB OF THE BINARY COUNTER.
616  01CA  A0 01             SUB     #1      LOBIN = LOBIN - 1 ; BORROW -> C,CCR
617  01CC  B7 11             STA     LOBIN
618                  ***
619  01CE  B6 10             LDA     HIBIN
620  01D0  A2 00             SBC     #0      SUBTRACT THE CARRY FROM THE HIGH BYTE.
621  01D2  B7 10             STA     HIBIN
622                  ***
623  01D4  B6 10             LDA     HIBIN
624  01D6  2A 1B             BPL     CCWPOS      IF HIBIN .GE. 0 , WE'RE ROTATING CCW TOWARD
625                  ***                         THE ORIGIN.
626  01D8  A1 FF             CMP     #-1     --
627  01DA  26 10             BNE     CCWNEG  --  ELSE IF BINCT .NE. -1 ,
628  01DC  B6 11             LDA     LOBIN     --WE'RE ROTATING CCW AWAY
629  01DE  A1 FF             CMP     #-1     --  FROM THE ORIGIN.
630  01E0  26 0A             BNE     CCWNEG  --
631  01E2  14 20             BSET    NEGTIV,STAT ELSE, WE'VE GONE THROUGH ORIGIN IN CCW
632                  ***                         DIRECTION.  SET NEGATIVE SIGN.
633                  ***                         AND SET ALL COUNTERS APPROPRIATELY.
634  01E4  AE 12             LDX     #PTR3   --
635  01E6  7F        CLREM   CLR     ,X      --
636  01E7  5C                INCX            -- RESET ALL COUNTERS AND DEGRES TO ZERO.
637  01E8  A3 19             CPX     #THOUTH --
638  01EA  23 FA             BLS     CLREM   --
639                  ***
640  01EC  AD 1F     CCWNEG  BSR     ADDBCD
641  01EE  CD 02 32          JSR     INCPOS
642  01F1  20 06             BRA     UPOUT
643                  ***
644  01F3  CD 02 7A  CCWPOS  JSR     SUBBCD
645  01F6  CD 02 9A          JSR     DECPOS
646                  ***
647                  ***
648                  **
649                  ********************************************************************************
650                  **
651                  ** OUTPUT ROUTINE. ROUTINE TO PRINT DATA TO THE OUTPUT PORTS.  BY
652                  **       CALLING THE APPROPRIATE SUBROUTINE.('OUTCT' TO OUTPUT THE
653                  **       THE COUNT AND 'OUTPOS' TO OUTPUT THE POSITION).
654                  **
655       01F9       UPOUT   EQU     $
```

```
                        **
01F9  04 20 04          MINUS   BRSET   NEGTIV,STAT,MINUS  --
01FC  1C 01                     BSET    POSTIV,PORTB         --
01FE  20 02                     BRA     DISCHK                 --
                        **                                     --    SET THE NEGATIVE SIGN
0200  1D 01             MINUS   BCLR    POSTIV,PORTB         --      APPROPRIATELY.
                        **
0202  09 20 04          DISCHK  BRCLR   POSCT,STAT,PUTCT
                        **
0205  CD 02 E5                 JSR     OUTPOS
0208  80                       RTI
                        **
0209  CD 02 D8          PUTCT   JSR     OUTCT
020C  80                       RTI
                        **
                        ********************************************************************************
                        ***********  SUBROUTINE TO INCREMENT THE BCD COUNTER (BCDCT).  *********
                        ***
020D  B6 16             ADDBCD  LDA     CTPTR1
020F  AB 01                     ADD     #1
0211  A1 63                     CMP     #99     CTPTR > 99 ?
0213  23 05                     BLS     OK1     NO, WE'RE OK HERE. LOOK UP THE FIRST TWO DIGITS.
0215  A0 64                     SUB     #100    YES... MODIFY THE CTPTR,
0217  99                       SEC             SET THE CARRY, AND
0218  20 01                     BRA     OK1A    USE TABLE LOOK UP.
                        ***
021A  98                OK1     CLC                     NO CARRY EXISTS IF WE ENTER AT THIS POINT.
021B  B7 16             OK1A    STA     CTPTR1  --
021D  97                        TAX               --
                        ***                             -- LOOK UP THE TWO LEAST SIGNIFICANT DIGITS.
021E  D6 03 28                  LDA     TABLE,X   --
0221  B7 1B                     STA     TENONE  --
0223  24 0C                     BCC     NOMO            AND CONTINUE ONLY IF THERE WAS A CARRY.
                        ***                               .
                        ***
0225  B6 15                     LDA     CTPTR2
0227  A9 00                     ADC     #0      ADD THE CARRY.
0229  B7 15                     STA     CTPTR2  --
022B  97                        TAX               --
                        ***                             -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
022C  D6 03 28                  LDA     TABLE,X   --
022F  B7 1A                     STA     HUNDRD  --
                        ***
0231  81                NOMO    RTS
                        ***
                        ********************************************************************************
                        *******  SUBROUTINE TO INCREMENT THE POSITION COUNTER (DEGRES).  *******
                        ***
                        ****  FIRST CHECK TO SEE IF THE BINARY COUNTER HAS REACHED A MODULO 32
                        ****     NUMBER.
                        ****
0232  00 20 04          INCPOS  BRCLR   MOD_32,STAT,INC7   'MOD_32,STAT' SET ?
                        ****
0235  A6 08                     LDA     #8                    YES ...
0237  20 02                     BRA     INC             INCREMENT THE POSITION BY 0.008 DEGREES.
                        ****
0239  A6 07             INC7    LDA     #7                    NO ....
023B  B7 1D             INC     STA     POSINC          INCREMENT THE POSITION BY 0.007 DEGREES.
                        ****
                        **** ROUTINE TO INCREMENT THE POSITION COUNTER ,'DEGREES',BY A
                        ****     PREDETERMINED AMOUNT, 'POSINC'.
                        ****
023D  B6 14                     LDA     PTR1
023F  BB 1D                     ADD     POSINC
0241  A1 09                     CMP     #9      PTR1 > 9 ?
0243  23 05                     BLS     OK3     NO, WE'RE OK HERE.  LOOK UP THE FIRST DIGIT.
```

```
722  0245  A0 0A          SUB   #10     YES... MODIFY THE CTPTR,
723  0247  99             SEC           SET THE CARRY, AND
724  0248  20 01          BRA   OK3A    USE TABLE LOOK UP.
725                ****
726  024A  98       OK3   CLC                 NO CARRY EXISTS IF WE ENTER AT THIS POINT.
727  024B  B7 14    OK3A  STA   PTR1    --
728  024D  97             TAX             --
729                ****                    -- LOOK UP THE LEAST SIGNIFICANT DIGIT.
730  024E  D6 03 28       LDA   TABLE,X   --
731  0251  B7 19          STA   TWOUTH    --
732  0253  24 24          BCC   DONE            AND CONTINUE ONLY IF THERE WAS A CARRY.
733                ****
734                ****
735  0255  B6 13          LDA   PTR2
736  0257  A9 00          ADC   #0      ADD THE CARRY.
737  0259  A1 63          CMP   #99     PTR2 > 99 ?
738  025B  23 05          BLS   OK4     NO, WE'RE OK HERE. LOOK UP THE NEXT TWO DIGITS.
739  025D  A0 64          SUB   #100    YES...MODIFY THE CTPTR,
740  025F  99             SEC           SET THE CARRY,
741  0260  20 01          BRA   OK4A    AND USE TABLE LOOK UP.
742                ****
743  0262  98       OK4   CLC                 NO CARRY EXISTS IF WE ENTER AT THIS POINT.
744  0263  B7 13    OK4A  STA   PTR2    --
745  0265  97             TAX             --
746                ****                    -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
747  0266  D6 03 28       LDA   TABLE,X   --
748  0269  B7 18          STA   HUNDTH    --
749  026B  24 0C          BCC   DONE            AND CONTINUE ONLY IF THERE WAS A CARRY.
750                ****
751                ****
752  026D  B6 12          LDA   PTR3
753  026F  A9 00          ADC   #0      ADD THE CARRY.
754  0271  B7 12          STA   PTR3    --
755  0273  97             TAX             --
756                ****                    -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
757  0274  D6 03 28       LDA   TABLE,X   --
758  0277  B7 17          STA   ONEDEG    --
759                ****
760  0279  81       DONE  RTS
761                ***
762                **
763                ********************************************************************************
764                ********  SUBROUTINE TO DECREMENT THE BCD COUNTER (BCDCT).  ************
765                ***
766  027A  B6 16    SUBBCD LDA   CTPTR1
767  027C  A0 01          SUB   #1      CTPTR > 99 ?
768  027E  24 03          BCC   OK6     NO, WE'RE OK HERE. LOOK UP THE FIRST TWO DIGITS.
769  0280  AB 64          ADD   #100    YES, MODIFY THE CTPTR, AND
770  0282  99             SEC           GENERATE A BORROW.
771                ***
772  0283  B7 16    OK6   STA   CTPTR1    --
773  0285  97             TAX             --
774                ***                     -- LOOK UP THE TWO LEAST SIGNIFICANT DIGITS.
775  0286  D6 03 28       LDA   TABLE,X   --
776  0289  B7 1B          STA   TENONE    --
777  028B  24 0C          BCC   COMPLT        AND CONTINUE ONLY IF THERE WAS A BORROW.
778                ***
779                ***
780  028D  B6 15          LDA   CTPTR2
781  028F  A2 00          SBC   #0      SUBTRACT THE CARRY. CTPTR > 99 ?
782  0291  B7 15          STA   CTPTR2    --
783  0293  97             TAX             --
784                ***                     -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
785  0294  D6 03 28       LDA   TABLE,X   --
786  0297  B7 1A          STA   HUNDRD    --
787                ***
```

```
0299  81          COMPLT  RTS
                  ***
                  ********************************************************************************
                  *******  SUBROUTINE TO DECREMENT THE POSITION COUNTER (DEGRES).  *******
                  ***
                  ****  FIRST CHECK TO SEE IF THE BINARY COUNTER HAS REACHED A MODULO 32
                  ****     NUMBER.
                  ****
029A  0D 20 04    DECPOS  BRCLR   MOD_32,STAT,DEC7        'MOD_32,STAT' SET ?
                  ****
029D  A6 08               LDA     #8                      YES...
029F  20 02               BRA     DEC             DECREMENT THE POSITION BY 0.008 DEGREES.
                  ****
02A1  A6 07       DEC7    LDA     #7                      NO ....
02A3  B7 1D       DEC     STA     POSINC          DECREMENT THE POSITION BY 0.007 DEGREES.
                  ****
                  **** ROUTINE TO DECREMENT THE POSITION COUNTER ,'DEGREES',BY A
                  ****     PREDETERMINED AMOUNT, 'POSINC'.
                  ****
02A5  B6 14               LDA     PTR1
02A7  B0 1D               SUB     POSINC  PTR1 < 0 ?
02A9  24 03               BCC     OK8     NO, WE'RE OK HERE. LOOK UP THE FIRST DIGIT.
02AB  AB 0A               ADD     #10     YES, MODIFY THE CTPTR, AND
02AD  99                  SEC             GENERATE A BORROW.
                  ****
02AE  B7 14       OK8     STA     PTR1    --
02B0  97                  TAX               --
                  ****                        -- LOOK UP THE LEAST SIGNIFICANT DIGIT.
02B1  D6 03 28            LDA     TABLE,X   --
02B4  B7 19               STA     THOUTH  --
02B6  24 1F               BCC     DUNSUB        AND CONTINUE ONLY IF THERE WAS A BORROW.
                  ***
                  ***                             .
02B8  B6 13               LDA     PTR2
02BA  A2 00               SBC     #0      SUBTRACT THE BORROW.  PTR2 < 0 ?
02BC  24 03               BCC     OK9     NO, WE'RE OK HERE. LOOK UP THE NEXT TWO DIGITS.
02BE  AB 64               ADD     #100    YES, MODIFY THE CTPTR, AND
02C0  99                  SEC             GENERATE A BORROW.
                  ***
02C1  B7 13       OK9     STA     PTR2    --
02C3  97                  TAX               --
                  ***                         -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
02C4  D6 03 28            LDA     TABLE,X   --
02C7  B7 18               STA     HUNDTH  --
02C9  24 0C               BCC     DUNSUB         AND CONTINUE ONLY IF THERE WAS A CARRY.
                  ***
                  ***
02CB  B6 12               LDA     PTR3
02CD  A2 00               SBC     #0      SUBTRACT THE BORROW.
02CF  B7 12               STA     PTR3    --
02D1  97                  TAX               --
                  ****                        -- LOOK UP THE NEXT TWO DIGITS IN THE TABLE.
02D2  D6 03 28            LDA     TABLE,X   --
02D5  B7 17               STA     ONEDEG  --
                  ****
02D7  81          DUNSUB  RTS
                  ***
                  **
                  **
                  ********************************************************************************
                  **
                  ** OUTPUT COUNT (OUTCT).  SUBROUTINE TO MOVE THE CURRENT COUNT (BCDCT)
                  **       TO THE OUTPUT PORTS.  REMOVES THE DECIMAL POINT FROM THE
                  **       DISPLAY AND BLANKS ALL BUT THE LEAST SIGNIFICANT DIGIT. ALSO
                  **       SETS THE MINUS SIGN IF APPROPRIATE.
                  **
```

```
854             02D8            OUTCT   EQU     $
855                             **
856     02D8    B6 1B                   LDA     TENONE
857     02DA    B7 02                   STA     PORTC
858                             **
859     02DC    B6 1A                   LDA     HUNDRD
860     02DE    B7 00                   STA     PORTA
861                             **
862     02E0    1E 01                   BSET    DECPT,PORTB
863     02E2    19 01                   BCLR    BLANK,PORTB
864                             **
865     02E4    81                      RTS
866                             **
867                             ********************************************************************************
868                             **
869                             ** OUTPUT POSITION (OUTPOS).  SUBROUTINE TO MOVE THE CURRENT POSITION
870                             **       COUNT (BCDCT) TO THE OUTPUT PORTS.  THE DECIMAL POINT IS
871                             **       DISPLAYED , AND ONLY THE MOST SIGNIFICANT DIGIT IS BLANKED.
872                             **
873             02E5            OUTPOS  EQU     $
874                             **
875     02E5    B6 19                   LDA     THOUTH  --
876     02E7    A1 05                   CMP     #5        --
877     02E9    25 2D                   BLO     TRUNC       --    IF 5 > 'THOUTH' SIMPLY TRUNCATE THE
878     02EB    B6 18                   LDA     HUNDTH        -- DISPLAY.  OTHERWISE...
879     02ED    A4 09                   AND     #9            -- IF THE LAST DIGIT ISN'T A NINE IT IS
880     02EF    A1 09                   CMP     #9            -- EASY TO ROUND UP.  JUST ADD A ONE.
881     02F1    26 1D                   BNE     DECIMAL       --
882     02F3    B6 18                   LDA     HUNDTH        -- BUT IF THE LAST DIGIT IS A NINE CHECK TO
883     02F5    A1 99                   CMP     #$99          -- SEE IF IT'S 99.  IF SO IT GETS GRIM.
884     02F7    27 04                   BEQ     UGLY          --
885     02F9    AB 07                   ADD     #7            -- IF THE NUMBER IS X9 AND X .NE. 9, THEN
886     02FB    20 1D                   BRA     PCOUT         -- JUST ADD SEVEN TO ROUND UP. DUE TO
887                             **                            -- HEXIDECIMAL.
888                             **                            -- IF THE LOW TWO DIGITS ARE BOTH NINES
889     02FD    A6 00           UGLY    LDA     #00         --   AND WE NEED TO ROUND UP....
890     02FF    B7 02                   STA     PORTC     --     MAKE THE LOW TWO DIGITS BOTH ZEROS
891     0301    99                      SEC             --     AND SET THE CARRY.
892                             **
893     0302    B6 17                   LDA     ONEDEG  --     CHECK THE LAST DIGIT AS BEFORE.
894     0304    A4 09                   AND     #9        --   IF USING THIS PORTION OF THE CODE
895     0306    A1 09                   CMP     #9          -- THERE HAD TO BE A CARRY.
896     0308    26 13                   BNE     NEXT          -- IF THE LAST DIGIT IS A NINE DO THE
897     030A    B6 17                   LDA     ONEDEG        -- CARRY HERE.  IF NOT USE THE ADC
898     030C    AB 07                   ADD     #7          --   INSTRUCTION TO TAKE CARE OF IT
899     030E    20 11                   BRA     PAOUT     --     BELOW.
900     0310    B6 18           DECIMAL LDA     HUNDTH --
901     0312    AB 01                   ADD     #1        --
902                             **                          --  THIS IS ALL THAT NEEDS TO BE DONE IF
903     0314    B7 02                   STA     PORTC     --     THE LAST DIGIT IS NOT A NINE.
904     0316    20 05                   BRA     NEXT    --
905                             **
906     0318    B6 18           TRUNC   LDA     HUNDTH  --
907     031A    B7 02           PCOUT   STA     PORTC     --  AND IF THERE IS NO CARRY IT'S EVEN EASIER.
908     031C    98                      CLC             --
909                             **
910     031D    B6 17           NEXT    LDA     ONEDEG
911     031F    A9 00                   ADC     #0
912     0321    B7 00           PAOUT   STA     PORTA
913                             **
914     0323    18 01                   BSET    BLANK,PORTB
915     0325    1F 01                   BCLR    DECPT,PORTB
916                             **
917     0327    81                      RTS
918                             **
919                             ********************************************************************************
```

```
920                                 **
921                                 ** SET UP THE TABLE TO BE USED WITH BCD INCREMENT/DECREMENT ROUTINES.
922                                 **
923                                 *       ENDS
924                                 *       DATA
925  0328  00 01 02 03 04  TABLE    FCB     $00,$01,$02,$03,$04,$05,$06,$07,$08,$09
     032D  05 06 07 08 09
926  0332  10 11 12 13 14           FCB     $10,$11,$12,$13,$14,$15,$16,$17,$18,$19
     0337  15 16 17 18 19
927  033C  20 21 22 23 24           FCB     $20,$21,$22,$23,$24,$25,$26,$27,$28,$29
     0341  25 26 27 28 29
928  0346  30 31 32 33 34           FCB     $30,$31,$32,$33,$34,$35,$36,$37,$38,$39
     0348  35 36 37 38 39
929  0350  40 41 42 43 44           FCB     $40,$41,$42,$43,$44,$45,$46,$47,$48,$49
     0355  45 46 47 48 49
930  035A  50 51 52 53 54           FCB     $50,$51,$52,$53,$54,$55,$56,$57,$58,$59
     035F  55 56 57 58 59
931  0364  60 61 62 63 64           FCB     $60,$61,$62,$63,$64,$65,$66,$67,$68,$69
     0369  65 66 67 68 69
932  036E  70 71 72 73 74           FCB     $70,$71,$72,$73,$74,$75,$76,$77,$78,$79
     0373  75 76 77 78 79
933  0378  80 81 82 83 84           FCB     $80,$81,$82,$83,$84,$85,$86,$87,$88,$89
     037D  85 86 87 88 89
934  0382  90 91 92 93 94           FCB     $90,$91,$92,$93,$94,$95,$96,$97,$98,$99
     0387  95 96 97 98 99
935                                 *       ENDS
936                                 *       CODE
937                                 ********************************************************************************
938                                 **
939                                 **                      SET UP MASK OPTION REGISTER.
940                                 **
941  038C                           ABSOLUTE        JUST TO ENSURE THAT THE INTERRUPT VECTORS
942                                 **              ARE CORECTLY LOCATED.
943                                 **
944                                 **
945  0F38                           ORG     MOR
946  0F38  07                       FCB     #BIT2+BIT1+BIT0
947                                 **
948                                 **      COMMENTS:
949                                 **      BIT 7   CLOCK SOURCE  0 = CRYSTAL.
950                                 **      BIT 6   TIMER OPTION  0 = INTERNAL.
951                                 **      BIT 5   TIMER/CLOCK SOURCE  0 = INTERNAL.
952                                 **      BIT 4   NOT USED.
953                                 **      BIT 3   NOT USED.
954                                 **      BIT 2   SET  -
955                                 **      BIT 1   SET    -  PRESCALE SELECT  111 => 128
956                                 **      BIT 0   SET  -
957                                 **
958                                 **
959                                 ********************************************************************************
960                                 **
961                                 **                       ASSIGN INTERRUPT VECTORS.
962                                 **
963  0FF8                           ORG     INTRPT
964                                 **
965  0FF8  0139                     FDB     BLINK   TIMER/INT2 INTERRUPT VECTOR.
966  0FFA  0158                     FDB     COUNT   EXTERNAL INTERRUPT VECTOR.
967  0FFC  0158                     FDB     COUNT   SOFTWARE INTERRUPT VECTOR, NOT USED.
968  0FFE  0080                     FDB     RESTRT  RESET VECTOR.
969                                 **
970                                 **
971                                 **
972                                 ENDS
973  1000                           END

        Lines Assembled :  973           Assembly Errors :  0
```

LIST OF REFERENCES

1. Ayers, Gary Robert, *Calibration and Initialization of the NPS Modified Infrared Search and Target Designation (IRSTD) System*, Master's Thesis, Naval Postgraduate School, Monterey, CA, December 1987.

2. Norton, Harry N., *Sensor and Analyzer Handbook*, Prentice Hall Inc., 1982.

3. Herceg, Edward E., *Handbook of Measurement and Control*, Schaevitz Engineering, 1972.

4. Lenk, John D., *Handbook of Controls and Instrumentation*, Prentice Hall Inc., 1980.

5. Farrand Controls, *INDUCTOSYN Precision Linear and Rotary Position Transducers*, pp. 2-6, undated.

6. Doebelin, Ernest O., *Measurement Systems, Application and Design*, Mc Graw-Hill Book Co., 1976.

7. Hewlett Packard, *Application Note 1011, Design and Operational Considerations for the HEDS-5000 and HEDS-6000 Incremental Optical Shaft Encoders*, undated.

8. *Installation and Operation Manual, PT1250DC Heavy Duty Pan/Tilt*, p. 12, PELCO Sales Co., 1988.

9. Hewlett Packard, *56 mm Diameter Two and Three Channel Incremental Optical Encoder Kit*, Technical Data, January 1984.

10. Motorola Semiconductors, *MC68705U3 Advance Information*, Motorola Inc., 1981.

11. National Semiconductor Corporation, *Logic Databook*, Vol. 2, 1984.

12. Terman, F.W., *Latches and Flip Flops*, Lab Notes for EC2820 (Digital Circuits), pp. 3,4, Naval Postgraduate School, undated.

13. Taub, Herbert, *Digital Circuits and Microprocessors*, pp. 130-135, McGraw-Hill Book Co., 1982.

14. Motorola, *M6805 HMOS, M146805 CMOS Family, Microcomputer/Microprocessor User's Manual*, 2nd ed., Prentice-Hall Inc., 1983.

15. *Installation and Operation Manual, Motorized Zoom Lenses*, p. 6, PELCO Sales Co., undated.

16. National Semiconductor Corporation, *Linear Databook*, pp. (1-181)-(1-183), 1982.

17. 2500 A.D. Software, *6805 Macro Assembler*, Version 4.02, pp. (1-1)-(2-38), undated.

18. Motorola, *M68705EVM Evaluation Module User's Manual (M68705EVM/AD4)*, 4th ed., Motorola Inc., 1987.

19. Houghton, P.S., *Gears; Spur, Helical, Bevel, Internal, Epicyclic, and Worm*, 3rd ed., pp. 1.4,1.5, Technical Press, 1970.

20. Peebles, Peyton Z., Jr., *Probability, Random Variables, and Random Signals*, pp. 121-122, 2nd ed., McGraw-Hill Book Co., 1987.

21. Lloyd, Scot D., *An Autopilot Design for the United States Marine Corps' Airborne Remotely Operated Device*, Master's Thesis, Naval Postgraduate School, Monterey, CA, September 1987.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Commandant of the Marine Corps Code TE 06 Headquarters, U.S. Marine Corps Washington, D.C. 20380-0001	1
4.	Professor A. W. Cooper, Code 61Cr Department of Physics Naval Postgraduate School Monterey, CA 93943	10
5.	Department Chairman, Code 62 Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943	1
6.	Professor H. A. Titus, Code 62Ts Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943	2
7.	Professor G. J. Thaler, Code 62Tr Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943	1
8.	Captain Peter D. Lloyd 3719 Frostwood Rd. Knoxville, TN 37921	5